Texture Analysis and Synthesis using the Multiresolution Fourier Transform

Tao I Hsu B.Sc. MSEE

A thesis submitted to
The University of Warwick
for the degree of
Doctor of Philosophy

September 1994

Texture Synthesis and Analysis using the Multiresolution Fourier Transform

Tao I Hsu B.Sc. MSEE.

A thesis submitted to
The University of Warwick
for the degree of
Doctor of Philosophy

September 1994

Summary

In this thesis, a new frequency domain approach to analysis of texture is presented, in which both the statistical and structural aspects of the problem are combined in a unified framework, the *Multiresolution Fourier Transform* (MFT). The analysis scheme consists of two main components: texture synthesis and texture segmentation.

The synthesis method works by identifying, for pairs of texture 'patches' of a given size, the affine co-ordinate transformation which gives the best match between them. This allows the analysis to take account of the geometric warping which is typically found in images of natural textures. By variation of scale, using the MFT, it is possible to identify the scale of the texels giving the best results in the matching process. The technique has the potential to deal effectively with textures having varying amounts of structure and can be used both for segmentation and resynthesis of textures from a single prototype block.

The texture segmentation makes use of the localisation properties of the MFT to detect texture boundaries using the MFT coefficient magnitudes, which incorporate both boundary and region information, in order to place texture boundaries accurately. A segmentation algorithm is described, starting with pre-smoothing to reduce texture fluctuations followed by an edge detection based on a Sobel operator to give an initial texture boundary estimate. Both boundary enhancement and region averaging are accomplished by adopting a 'link probability function' to introduce dependence on neighbouring data, allowing the boundary to be refined successively to achieve segmentation. The method effectively uses the spatial consistency of boundary estimates at larger scales to propagate more reliable information across scales to improve the accuracy of the boundary estimate.

Experimental results are presented for a number of synthetic and natural images having varying degrees of structural regularity and show the efficacy of the methods.

Key Words:
Texture synthesis, texture segmentation, texture scale, multiresolution, affine transform

Contents

List of Figures

List of Tables

Acknowledgements

This work was supported by the Government of the Republic of China on Taiwan and conducted within the Image and Signal Processing Research Group in the Department of Computer Science at Warwick University.

I would like to thank all the staff of the Computer Science Department. In particular, thanks go to all my friends and colleagues, past and present, of the Image and Signal Processing Group at Warwick: Abhir Bhalerao, Simon Clippingdale, Nicola Cross, Andrew Davies, Wooi Boon Goh, Andy King, Peter Meulemans, Edward Pearson, Hugh Scott, Tim Shuttleworth, Martin Todd, and Horn-Chang Yang. They have made numerous contributions and have provided a stimulating environment in which to work. Thanks also to Jeff Smith for providing essential software support.

I would like especially to thank Dr. Andrew Calway for his enthusiasm and stimulating discussions regarding to the work of texture synthesis.

I am particularly indebted to my supervisor Dr. Roland Wilson, without whose ideas, enthusiasm and profound expertise in the subject this work would not have been possible.

Declaration

I declare that, except where specifically acknowledged, the material contained in this thesis is my own work and that it has neither been previously published nor submitted elsewhere for the purpose of obtaining an academic degree.

Tao I Hsu

This work is dedicated to my wife Yu Chen.

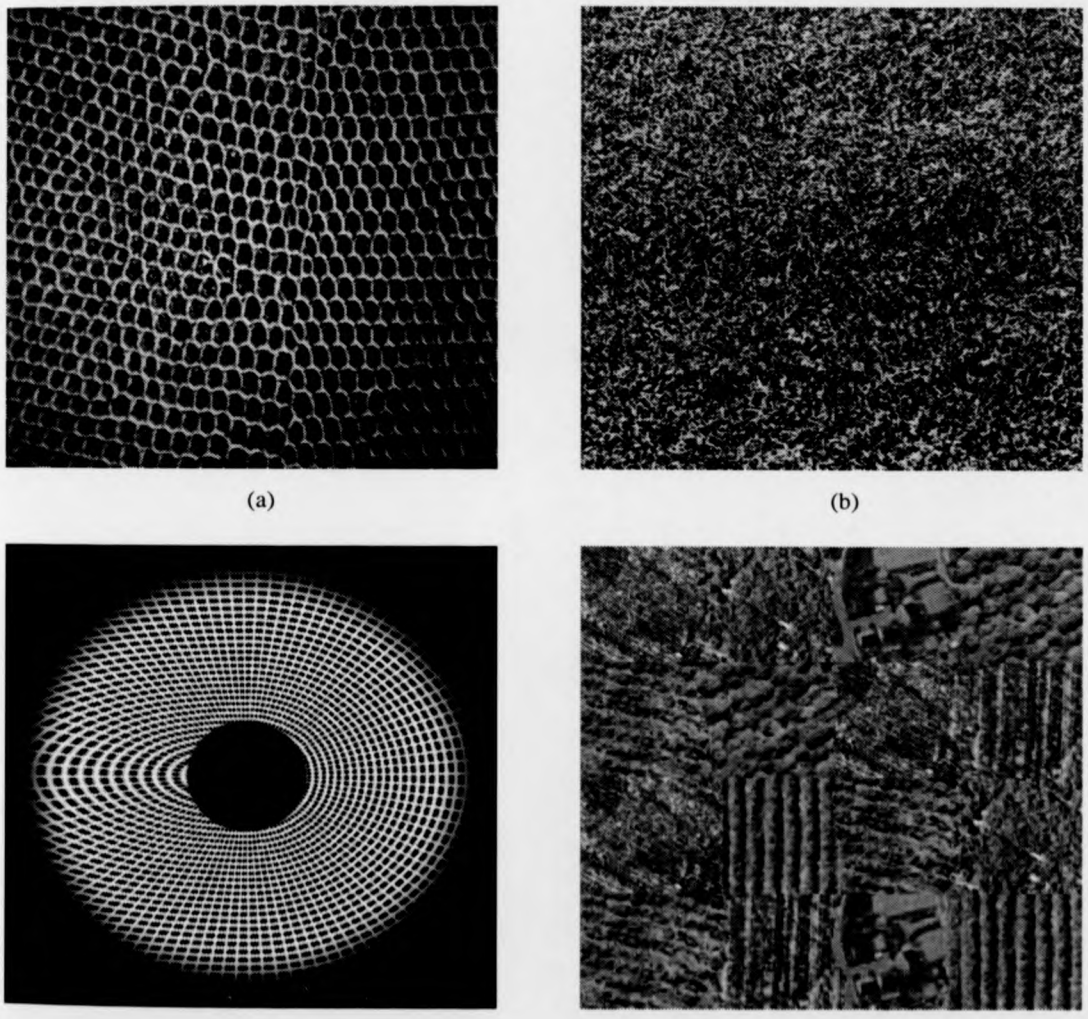
Chapter 1

Introduction

1.1 Introductory Remarks

Texture is a ubiquitous property of visible surfaces, but there is no formal or precise definition of texture. The term 'texture' generally refers to repetition of some elemental unit over a region which is large compared to that of the elemental unit [54]. This element contains several pixels, whose arrangement may follow certain rules, which may be periodic, quasi-periodic or random. A large class of textures can be covered by such a definition, ranging from structural texture (reptile skin) and random texture (grass) to crop imagery, as shown in figure 1.1.

Texture is one of the important characteristics which exists for analysis in many types of natural images, from satellite imagery in remote sensing, to microscopic images of medical samples, to outdoor scenes and object surfaces. The problem of texture description centres around both what precisely is meant by 'texture', for there is a large number of attributes of texture that may be defined, and how such textured regions should be generated and analysed.



(a) (b)

(c) (d)

Figure 1.1: Different examples of textured image.

1.1.1 Visual Texture Perception

Visual textures have great variability. How human beings perceive texture and what the crucial factors are that affect the discrimination of two texture fields are two important issues in deciding the characteristics of what we see. A large amount of experimental evidence has shown that for detection of visual stimuli, the visual system behaves as if it were composed of a number of independent channels, each sensitive to a limited range of spatial frequencies and orientations [62] [47]. From the psychological and physiological view [87], an important determinant in the detection of the stimulus by the human visual system is its spatial frequency content.

Two hypothesised mechanisms of the human visual system for texture perception are 'texton' detection and texture filtering. The first class of models reported by Julesz et al [61] [60] regard the fundamental texture elements as textons, which are used to describe features of individual sub-patterns that render texture pairs with identical second order statistics discriminable. Discrimination is accomplished by the comparison of the relative texton densities, which is claimed to be consistent with the visual discrimination of textures. However, the difficulty in using this approach to the segregation of textures is that no suitable mechanism was proposed for local extraction of textons or the texture boundary.

Among models of texture filtering, Caelli [15] advocated a Fourier model which is able to simulate the human visual system on the basis of mechanisms which decompose an image into its frequency components, including both amplitude and phase coding. A set of 'clam shell' filters centering on different parts of the Fourier domain and of different orientation selectivity were used by Caelli to discriminate texture

fields. While these efforts have demonstrated that a filtering approach can explain some phenomena that are not consistent with texton theory, a complete model has not yet been presented. Such a model should be able to unify the solution of the conflicting problems of determining local textural structures (texture element, texture boundaries) and identifying the spatial extent of the textured regions contributing significant spectral information.

1.1.2 Texture Analysis Problems

Most texture research can be characterised by the underlying assumptions made about the texture formation process. These assumptions depend primarily on the type of textures to be considered in the study. This leads to classification of the approaches for texture analysis into either statistical approaches, considering the texture as defined by a set of statistics extracted from a large ensemble of local image measurements, or structural approaches, using the concept that a textured image is composed of texture elements.

Statistical Approaches

The conventional statistical methods for texture analysis summarise the gray level distribution into a finite number of statistics, among them the well-known co-occurrence matrices [51], [110]. These methods are based on the estimation of the second-order joint conditional probability density function, $f(i,j|d,\theta)$. Each $f(i,j|d,\theta)$ is the probability of going from gray level i to gray level j, given the intersample spacing distance d, and the direction given by the angle θ. The estimated values, $x_\theta(i,j|d)$, can be written in matrix form, the so-called co-occurrence matrix. Various versions of co-

occurrence matrices have been developed and their capability has been demonstrated [23] [46]. In these methods, textural structures which can be described within a neighbourhood are naturally limited to those which are observable within the size of the neighbourhood. Thus a feature based on measurements within a neighbourhood, fixed in size, has poor discrimination power when applied to a texture not observable within the neighbourhood because it has the wrong scale. But in general the scale information is not available. This is a fundamental limitation of such methods.

Alternatively, a parametric model may be used for statistical modelling. The most common stochastic models of texture are the autoregressive [58], or more generally Markovian Random Field (MRF) models [25] [44]. Chellappa and Kashyap [18], suggested the use of 2-D autoregressive, noncausal models for textures. In this model, the gray level of a pixel is characterised as a linear combination of the gray levels at neighbouring pixels and an additive linear combination of some noise field values. The model parameters are estimated by approximating a maximum likelihood solution for various possible neighbourhoods. One of these models is chosen, using a Bayesian decision rule, to approximate the original noncausal model and then to synthesise images. Chen [19] proposed a maximum entropy power spectrum estimation method, which involves a parameteric way of estimating spectrum based on an autoregressive model. These methods are widely used but are of limited applicability to images because the causality restriction is significant: the autoregressive model is limited by its inherent directionality, which introduces a degree of anisotropy.

An edge-based segmentation technique based on features derived from a simultaneous autoregressive (SAR) random field was described by Khotanzad and Chen [2]. A grid of overlapping windows is used to partition the image of interest. Within each

of the windows, the least squares estimates of the SAR model parameters are used as texture features. Texture edges in each window are placed at positions where sudden changes in the features of a neighbouring window occur. Enhancement of individual feature edges is carried out via a block-based 3×3 Sobel operator to produce an initial edge map. The enhanced texture changes are combined to construct a 'textural measure' and thresholded to give a texture edge. This texture edge is further thinned to give a one-pixel wide skeleton. It should be noticed that the accuracy of the estimated edges is determined by the window, due to the use of block-based edge detection. Although the result is thinned, significant misplacement may still occur when an inappropriate scale of the window is used.

Stochastic image models based on the Gibbs distribution [45] are of great current interest. In [31], Derin and Elliot use the Gibbs distribution for texture segmentation via a dynamic programming method and maximum *a posteriori* (MAP) criterion which allows the use of standard least squares techniques iteratively. This work is applicable only to noise-free textured images consisting of known textures and with known Gibbs distribution parameters. This technique also requires a small enough neighbourhood, so that equations can be determined for obtaining the least squares solution.

Texture boundary detection incorporated by the introduction of a binary valued line process described by Geman et al [44] is another notable use of this approach. Two types of labels, partition and boundary labels, are used in a region partition process and boundary detection respectively. The local blocks of the image in the region process are augmented by line sites which are placed between each vertical and horizontal pair of blocks. The configuration space of any neighbourhood is extended to

include all possible pairs of allowable block and line states. The distance between pairs of neighbours is measured by the Kolmogorov-Smirnov distance and this is used to detect discontinuities. Adopting the Gibbs representation, a form of energy function is defined to place the boundary between regions and prohibit certain configurations that are false edges or very small regions, resulting in a reduction of the number of model parameters. Although the results shown are improved by introducing the line process, the size of the line process neighbourhood is artificially set by the region assignment process. As noticed in [1], in this modified MRF model, the model parameters cannot be estimated by standard methods, such as maximum likelihood estimation, because the solution of the likelihood equation compounds the problem of model parameter identification, resulting in computational intractability.

Elfadel and Picard [36] examined a method which describes an explicit relationship between co-occurrence matrices and Gibbs random fields. An intermediate representation including an 'aura set' and 'aura measure' is developed to establish the connection with morphological dilation combined with co-occurrences. Based on the concepts of the Gibbs energy and MAP estimation, the results of their method show that their work is still in its initial stages.

More recently, the Gaussian Markov Random Field (GMRF) [22], [21] and [17] was applied in unsupervised texture segmentation. An image of interest is divided into disjoint regions; over these regions the GMRF parameters are estimated. The regions are merged via clustering based on the normalised Euclidean distance between their parameter vectors. Following recomputing of the parameters over the merged regions, a refined segmentation is carried out via pixel-based segmentation. The boundary precision obtained by this method is quite good.

One difficulty for these methods is that the texture elements in the stochastic process are not highly structured and large data sets may be required to get good parameter estimates [83] [38]. In general, statistical methods are well suited to textures which are random in nature, such as sand, grass and perhaps water, whilst textures such as tilings of the plane, brick wall and reptile skin are not appropriate for this type of model.

Structural Approaches

Unlike the stochastic models, structural approaches have not received much attention recently [51] [53]. They include the development of a compact structural description of texture by the minimum number of parameters needed to identify the placement rule of the texture elements [82]. In general, these methods first locate texture elements, then analyse spatial arrangements. In natural images, texture fields may be degraded during imaging and obscured by noise and blur. Even with complete knowledge of texture types, it is more difficult to isolate the texture elements than it is to generate them.

Matsuyama et al [71] resolved this difficulty by first estimating a pair of spatial vectors from the Fourier energy spectrum representing the placement rule of the texture elements. Then, following inverse Fourier transformation of the estimated vectors, texture elements are extracted by a region growing method. The basic disadvantage of this method is its preliminary assumption of the completely periodic structure of the texture field. The synthesis procedure is performed by placing 'averaged' texture elements according to the estimated placement rule, resulting in artificial looking textures.

Similarly, the method described in Volet and Kunt [97] extracts the placement rule first, based on a pair of linearly independent vectors derived from the local autocorrelation function of the image. With the extracted placement rule, the texture is transformed to a 'normalised' co-ordinate space via an affine transform and the texture element is then extracted. A global average texture element is obtained by averaging the normalised local texture elements and used with geometric warping in order to reconstruct the image. McColl has applied this idea in texture coding [72].

Vilnrotter et al [96] examine the statistics of edge features to control texture element identification and isolation. From the isolated texture elements, a set of placement rules are computed. The reconstruction is performed by combining these into a complete description. As in earlier structural methods, the results are acceptable only in the case of periodic textures. These structural methods have generally assumed that the textures exhibit some degree of regularity, which is violated in many textures [53].

Summary

Both of these types of methods depend highly on the texture type, which restricts their applicability. Thus, while statistical techniques are generally good for random textures and have poor performance on structural textures, the reverse is the case for the structural techniques. Ideally, a method which is capable of dealing with the full spectrum of texture types is required.

Having defined textured images as composed of texture elements and rules which describe the spatial arrangement of the texture elements, there are two basic questions in texture analysis: texture element estimation and description of the placement rules.

Texture element estimation is concerned with local property measurement. However, textures occurring in natural images generally show randomness or variability and exhibit fluctuations, ranging from the degradations occurring in the image formation process, surface irregularity and geometric warping caused by the 3-D to 2-D projection. Inevitably, these intrinsic properties affect the local texture measures used in analysis. Solutions often involve operations with local windows, which requires choosing the windowing function and window size. Choosing the right scale is a fundamental problem because texture is an area property [52]. There is seldom a systematic basis for choosing the window size. If the window is too big, for example, it may be difficult to identify region boundaries or adapt to variations within regions, while if it is too small it may give unsufficient discrimination of the textures.

Apart from the above problem, the windowing function prevents an image being considered at arbitrarily high resolution in both spatial and spatial frequency domains. This problem is a consequence of the *uncertainty principle* of signal analysis, which states that the size of the regions of support of a windowing function in space and frequency are inversely related - the smaller the window in frequency, the larger will it be in space and vice versa [75] [108]. Uncertainty is a problem for any window size used in spatial/spatial frequency representation. Daugman [28] shows that the 2-D Gabor function [40] can achieve simultaneous localisation in space and in spatial frequency to resolve the uncertainty problem. A number of researchers have used Gabor functions as filters to establish a scheme of multichannel filtering, which involves the decomposition of images into multiple feature images of narrow spatial frequency and orientation bandwidth [9] [59] [92] and [34]. Some issues exist in these approaches concerning the number and the central frequency of the Gabor filters used. More-

over, to tune each channel to a specific narrow band of spatial frequencies generally requires prior knowledge, which is unlikely to be available in the unsupervised mode of analysis.

In analysing the spatial arrangement of elements, it would be useful to find a method which is intermediate between the random placement of stochastic methods and the rigid rules of the simple structural approaches. This requires a method which has sufficient flexibility to allow for the sort of variation exhibited in the textures shown in figure 1.1 and in which the scale of the description can be matched to that of the texture elements.

Finally, segmentation of textured images requires location of the boundaries between the textured regions. The task of boundary detection between textured regions - texture segmentation - shares a common problem in vision, which Marr defines as 'knowing what is where by looking' [70]. 'What' is the element of texture, which establishes the class property of the perceived image, while 'where' is the position of the perceived texture in the spatial domain [108]. The main question in texture segmentation regarding the texture properties and positions is raised by the fact that identification of the texture type requires global information to yield a reliable description, which reduces the precision of the local boundary detection. Similarly, the texture boundary can be detected more accurately only at the cost of a loss of reliability of the classification. It implies that there is an incompatibility between texture class and position inherent in texture segmentation [108].

1.1.3 Aims of This Work

Central to the requirement of a model which can both locally detect texture elements and globally describe the density of the texture elements simultaneously, is the availability of a framework allowing the image to be considered at different scales and the information at these different scales to be combined or compared. A solution to this is to use a multiresolution approach (e.g. [85] [14] [109] [67] and [73]). A number of workers have employed such techniques in texture analysis [78] [91]. More recently, a classification scheme for texture analysis in the context of multiresolution imagery was described by Roan et al [84]. In this work, the statistical features were derived from the Fourier energy spectrum and co-occurrence matrices to study the relationship between feature properties and image resolution, which provides a useful approach in textured surface description. Unser and Eden [95] used the output of a filter bank to estimate local statistics of texture iteratively, giving a multiresolution sequence of feature planes. A feature reduction technique was proposed, allowing variety textures to be analysed in a single reduced component, but the problem of unsupervised determination of optimal window size remains unsolved. In [8], Bouman and Liu applied a causal Gaussian autoregressive model to describe the statistics of the textures. A coarse to fine segmentation was used in a multiresolution framework, requiring significantly less computation than earlier work using single resolution methods. Mao and Jain [69] suggested a multiresolution simultaneous autoregressive model (MR-SAR) for texture classification and segmentation. The method shows MR-SAR can achieve a substantial improvement over the single resolution SAR model. The multiresolution concept, which can give better performance and reduce computation, has attracted

researchers and will be discussed more in Chapter 2.

To summarise, the objectives of this work are:

1. To investigate the characteristics of texture features.

2. To devise a technique which allows for the modelling of global image structure in the same framework as that used for the local texture estimate, making computation efficient, as well as providing a consistent interface between local and global structure. This will allow analysis and synthesis of textures with varying amounts of structure.

3. To create a boundary detection algorithm, which combines information of both boundary position and texture properties at different scales and accommodates neighbourhood information to achieve segmentation of textured images.

1.2 Thesis Outline

A multiresolution approach to the analysis of texture is presented in this thesis. The multiresolution Fourier transform (MFT) is used as a framework to derive a robust algorithm which estimates textural features and detects texture boundaries over a range of spatial scales, based on local frequency domain properties.

In particular, in Chapter 2, the requirements of an image texture description will be discussed. The emphasis is placed on locality in representing the texture features. A number of existing models which might provide such description are reviewed. However, evidence will be given that these fail to provide sufficient locality in describing texture features. This leads to the modelling of the texture using the multiresolution Fourier transform (MFT).

Chapter 3 is devoted to considering the definition and implementation of the MFT. The MFT characteristics and properties will be discussed. Two different interpretations of the MFT suggest two corresponding implementations. The implementations can be accomplished efficiently using the FFT algorithm. Initial experiments and examples are presented for both synthetic and natural images.

Chapter 4 introduces a new texture 'synthesis and analysis' algorithm based on the MFT framework. This algorithm models textures by the use of affine co-ordinate transformation of a single, representative texture element. It extends the earlier work reported by Spann and Wilson [104] to model the structure in textures in a way which allows for variation of the texture structure. Results are presented of texture synthesis based on this approach.

In Chapter 5, a texture segmentation algorithm is described, which is designed using a cooperative algorithm within the MFT framework. The detection of texture boundaries uses a combination of boundary information and region properties. Results are presented that illustrate the effectiveness of the scheme.

The thesis is concluded in Chapter 6 with a synopsis and a discussion of some aspects of the work which could be extended or improved.

Chapter 2

Representations for Texture Analysis

2.1 Introduction

Texture can be used in image analysis in several ways, such as classification and discrimination of surface patterns, computation of the geometric structure of perceptual objects and segmentation of the scene into different homogeneous regions [108]. It was shown in Chapter 1 that the analysis of textures is one of the oldest problems in image processing, but continues to generate interest and a degree of controversy because of the limited success with which most of the methods of analysis have met [51] [46] [83].

The purpose of this chapter is to present and evaluate the characteristics of different texture analysis approaches which have been proposed, particularly the Fourier related approaches. To overcome limitations in those approaches, a general signal representation for texture analysis is set forth.

2.2 Requirements of a Texture Analysis Method:

The problem of obtaining useful texture measures is difficult because of the multiplicity of texture properties and their variation with parameters such as illumination, shading and surface reflectance. There is the question of finding an adequate description of texture - of selecting features which are sufficient to allow discrimination of the types of texture encounted in a given application. Furthermore, the perspective projection of 3-D textured surfaces into 2-D images causes distortions in the area, orientation, shape and coarseness of the texture elements, and many natural textured images exhibit a combination of regularity, such as approximate periodicity, and variation which is hard to model using either straightforward repetition or traditional statistical techniques [51, 25, 112].

In order to find an adequate description of texture, it is possible to formulate the requirements of a representation for texture analysis as linearity, invertibility, symmetry and locality:

2.2.1 Linearity

Linearity can provide the conditions to ensure the predictable response of the possible addition and filtering operations on the signal. If the descriptor is linear, the computation can be simplified and filtering operations can be more easily applied.

2.2.2 Invertibility

Image processing consists of a number of operations performed on the image. These operations transform the original image to another feature space and in this transformed domain the image is represented by some meaningful symbols. These oper-

ations usually cause a loss of information in passing from the original image to its feature-space representation. Very often, the dimensionality of the feature-space is lower than that of the image space, and only an approximation of the original image based on this partial information is reconstructed. An invertible operation preserves information, and in principle allows the original image to be recovered, while also providing an error measurement between the approximation and the original image.

2.2.3 Symmetry

The processes which produce the 2-D texture can in many cases be approximated by affine transforms. These processes include the biological or mechanical process by which the structure originated and the perspective transformation resulting from the projection onto 2-D. The 2-D affine transform is a linear co-ordinate transform followed by a shift and is given by

$$T(\vec{\xi}) = A\,\vec{\xi} + \vec{\gamma} \tag{2.1}$$

where A is a matrix dealing with the rotation, scaling and shear, $\vec{\gamma}$ is the translation and $\vec{\xi}$ is the co-ordinate. For example, in the reptile skin of figure 2.1 [11], the image can be interpreted as a texture element associated with a geometric placement rule which is approximately a 2-D affine transform. This geometric structure must be embedded in the descriptor, which is the essence of symmetry.

2.2.4 Locality

In general, images are so complex that they must be represented and analysed locally. There is substantial evidence that visual texture discrimination is a local process [66]. The ability to differentiate between two texture fields is based on the characteristics

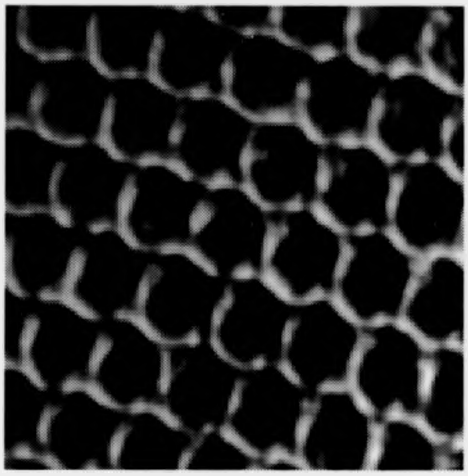


Figure 2.1: A reptile skin image illustrates the texture structure and the geometric transformation during the image generating process.

in relatively small neighbourhoods, rather than on the overall image characteristics and segmentation requires location of boundaries. Thus the representation should be local.

2.3 Approaches to Texture Analysis

In the previous section, it was established that some requirements of texture analysis method in order to find an adequate description of texture which is capable of analysing of varying amount of textural structure. Numerous approaches have been utilised in texture analysis, among which the co-occurrence matrix [51], gray level run lengths [42] and gray level difference methods [23] are well known (cf Chapter 1). Particularly, methods transforming the image into a corresponding domain wherein the analysis task is accomplished in the original domain and/or the corresponding

domain have received considerable attention in the literature [46] [83]. The purpose of this section is to discuss and evaluate their characteristics.

2.4 Fourier Based Texture Analysis

Many of the traditional texture features have explicit interpretations in the frequency domain such as 'fineness' corresponding to high frequency components, and 'regularity' corresponding to peaks in the spectrum. Physiological and psychological evidence suggest that the human visual system is capable of performing a form of spatial frequency analysis [29].

The chief properties of texture are periodicity and directionality, so that a suitable texture analysis description has to be able to characterise these two features [15]. In Fourier analysis, a 2-D signal $g(x, y)$ is decomposed into a linear combination of complex exponential functions of the form $e^{j(x\omega_x+y\omega_y)}$ and with coefficients given by

$$G(\omega_x, \omega_y) = \int\int_{-\infty}^{\infty} g(x, y)e^{-j(x\omega_x+y\omega_y)}dxdy \tag{2.2}$$

The complex exponential $e^{j(x\omega_x+y\omega_y)}$ has interesting properties. Consider the case that for a given frequency co-ordinate (ω_x, ω_y) and for some n

$$x\omega_x + y\omega_y = 2\pi n \tag{2.3}$$

then

$$e^{+j(x\omega_x+y\omega_y)} = e^{+j2\pi n} = 1 \tag{2.4}$$

In other words, the basis function with frequencies (ω_x, ω_y) is a wave in a specific orientation, with a real part as in figure 2.2. From (2.3) we know that

$$y = -\frac{\omega_x}{\omega_y}x + \frac{2\pi n}{\omega_y} \tag{2.5}$$

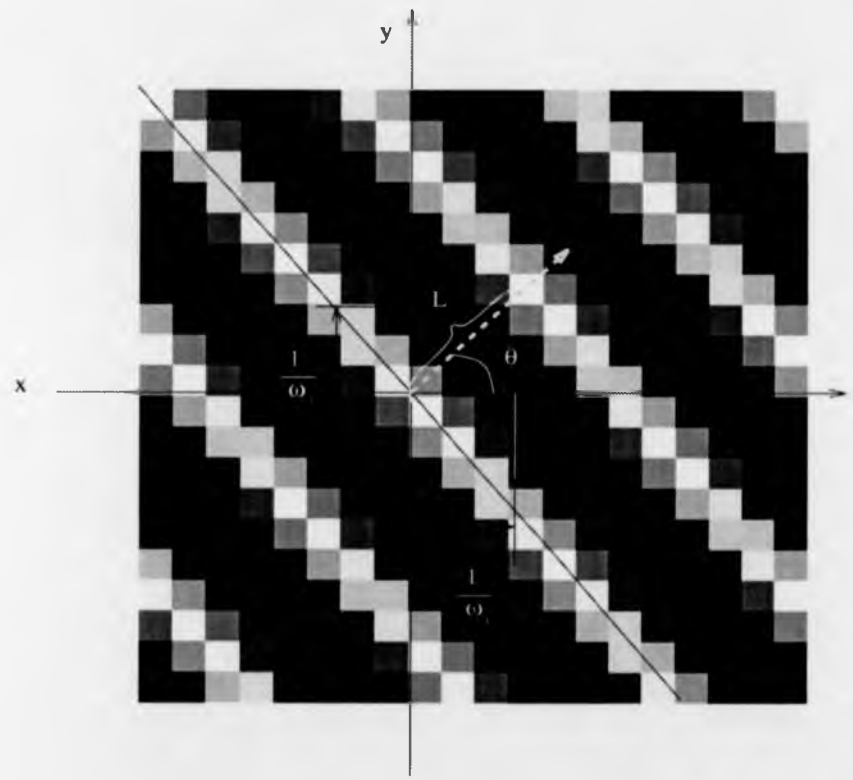


Figure 2.2: Real part of a Fourier basis function.

That is, the slope of each line is $-\frac{\omega_x}{\omega_y}$, and the normal to all of them is a line oriented at an angle $\theta = \arctan(\frac{\omega_y}{\omega_x})$. The spatial period (i.e. the distance between zero phase lines) is given by

$$L = \frac{1}{\sqrt{\omega_x^2 + \omega_y^2}} \tag{2.6}$$

Thus the Fourier transform is a natural choice for texture analysis, which underlies many of the methods described in the literature [4] [5] [37]. The Fourier transform defined in (2.2) leads to a rich mathematical structure associated with the transform operation. These mathematical properties such as linearity, invertibility and symmetry have ensured that the Fourier transform provides a general texture representation. In the aspect of locality, however, the Fourier transform has to be enhanced in order to become a general texture analysis tool. This will be addressed in the following sections.

In addition to the secure mathematical structure, the Fourier transform is convenient for handling noise. If there is noise within the textural image, the noise process will alter the image representation dramatically in the spatial domain, but uniformly in the frequency domain. Hence, frequency domain measures should be less sensitive to the noise than those in the spatial domain. Fourier based analysis is of most relevance to this work. Traditional Fourier based texture analysis uses the energy spectrum, which is the Fourier transform of the image autocorrelation function [75].

2.4.1 Autocorrelation Methods

Texture is an extended, spatially nonlocal or region property. Since texture is defined in relation to a neighbourhood, it is meaningless at an isolated pixel. The relationship between two pixels or regions is the significant feature in texture analysis. This

suggests the autocorrelation function $r(m,n)$ as a texture descriptor defined as

$$r_g(m,n) = \frac{1}{N^2} \sum_{k=0}^{N-1} \sum_{l=0}^{N-1} g(k,l)g(m+k,n+l) \tag{2.7}$$

where m and n are known as the 'lag'. For the sake of efficient computation, the autocorrelation may be calculated using Fourier transforms [75]. There are two types of information contained in $r_g(m,n)$. If the peak of the autocorrelation function falls off quickly as the lag departs from zero, this corresponds to a finer texture. A high-energy peak associated with $r_g(m,n)$ for nonzero lag also reveals a regularity in the spacing of texture elements in patterns such as wall paper. In terms of these two useful texture measurements, the autocorrelation is able to measure the directionality and coarseness of the texture.

The autocorrelation methods are based on spatial averages of second-order interactions. Methods which involve higher order statistics [41] require more memory space and are computationally expensive. The autocorrelation function, however, is not sufficient to describe a texture completely, since many natural textures have similar autocorrelation functions. Furthermore, its estimation often requires a large data window. In such cases, the description loses the locality of the texture identification and the computational cost increases dramatically.

2.4.2 Energy Spectrum Methods

Bajcsy [4] first used the Fourier energy spectrum to compute the sample energy spectrum with radial and wedge segments in analyzing texture. She also reported that phase information does not appear to yield useful texture measures. Rosenfeld et al., [99] used ring-shaped and wedge-shaped segments of the discrete Fourier energy

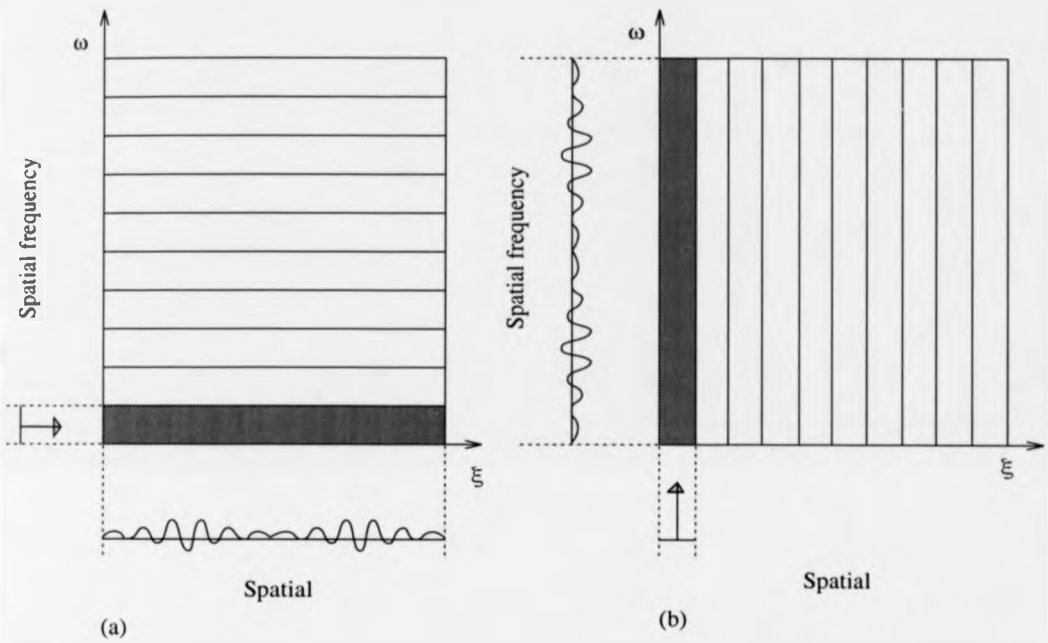


Figure 2.3: The Fourier transform is a global transform, (a) a single frequency component is a linear combination of spatial information across the entire image, and (b) a single spatial value is a linear combination of spatial frequency information across the entire frequency domain.

spectrum to obtain information about the size of texture elements and orientation of the texture image.

Such Fourier transform methods have met limited success in texture analysis. The most basic criticism of the frequency domain representation is that each frequency component contains global information from across the complete image and might neglect important local discriminating information about the texture (see figure 2.3). Instead of obtaining global information, the local properties of textures can be characterised by making use of a set of localised window functions over the textures. Each

localised window function highlights a particular property of the texture element or the response of the human visual system.

There are some difficulties associated with these measures:

1. The ring and wedge shaped functions used to compute the texture features were essentially a windowed Fourier transform (cf. section 2.5.1). Spatial locality is introduced by windowing the data prior to the integration of the energy of the transformed windowed data. However, this involves a loss in spatial frequency resolution, as the spectrum of the image is now convolved with that of the spatial window.

2. One consequence of windowing is that it destroys symmetry under co-ordinate transformations since windowing and co-ordinate transformations do not commute. That is, some features of an image within a window might disappear after co-ordinate transformation of that image. This raises the uncertainty problem of signal analysis. Uncertainty will be a problem for any window size used in spatial/spatial frequency representation (cf. section 2.5.1).

3. Coarseness in textural features is always scale dependent. This scale characteristic should be resolved in the feature descriptor to allow the image to be analysed at different scales. The selection of the appropriate processing scale is essentially context-dependent. Some of the texture element extraction has assumed that the processing scale is known a priori.

4. Feature information in both the spatial domain and frequency domain is required for segmentation tasks. This requires a careful choice of window function [108].

2.5 Spatial and Spatial Frequency Methods

Combined spatial and spatial frequency representations are useful in classifying non-stationary signals like textured images. Features which have identical or similar spatial or spectral characteristics may have very different characteristics in the complementary domain, which allow them to be distinguished. In addition, physiological and psychological experiments have shown that the human visual system processes visual information at different frequencies separately and is highly orientationally selective [37] [15]. For the purpose of texture segmentation, perceptual grouping is dependent on both the spatial and spectral structure. All these factors point to the need for high resolution and a conjoint representation of the spatial and spatial frequency domains. This conjoint representation is known as a phase-space representation [27].

Recently, there has been much work based on phase-space analysis in which methods have become ever more sophisticated. Representations that provide information about a texture locally over various scales have received considerable attention in the literature. The best known of these approaches fall broadly into four classes: windowed Fourier transform, scale-space transform, pyramid transform, and wavelet transform.

2.5.1 Windowed Fourier Transform

The windowed Fourier transform (WFT) is the best known of the methods which provide intermediate representations of the signal both in spatial and frequency domains. The windowed Fourier transform or short-time Fourier transform (STFT) has been used in 1-D signal processing for signal coding and recognition [81]. The 1-D

windowed Fourier transform of a discrete-time signal $x(\xi)$ is given by

$$\hat{x}(\xi, f) = \sum_{m=-\infty}^{\infty} w(\xi - m)x(m)e^{-j\frac{2\pi}{M}fm} \quad 0 \leq f \leq M \tag{2.8}$$

where $w(\xi)$ is a window function and '$\hat{}$' is used to denote the Fourier transform. The WFT can resolve nonstationary signals which have distinct frequency content at different instants into 'quasi-stationary' signals by windowing the input. The scale of the window will decide the corresponding frequency resolution. To increase the frequency resolution, one has to increase the size of the spatial window, which causes nonstationarities which might occur within this duration to be blurred: the spatial resolution decreases.

Ring and wedge shaped analysis windows discussed in section 2.4.2 are inherently a windowed Fourier transform, but suffer from infinite spread in the spatial domain because of the discontinuous windows used. One of the best known WFT methods in image analysis is the Gabor transform [40]. The standard Gabor representation is given by

$$\hat{x}(\xi, \omega) = \int_{-\infty}^{\infty} w(\xi - m)x(m)e^{-j\omega m}dm \tag{2.9}$$

where $w(\xi)$ is a windowing function, which in the original Gabor scheme is a Gaussian function.

Denoting the windowing function, shifted in space and spatial frequency, as

$$w_{\xi_i,\omega_i}(\xi) = e^{j\omega_i\xi}w(\xi - \xi_i) \tag{2.10}$$

gives its Fourier transform as

$$\hat{w}_{\xi_i,\omega_i}(\omega) = e^{-j\xi_i\omega}\hat{w}(\omega - \omega_i) \tag{2.11}$$

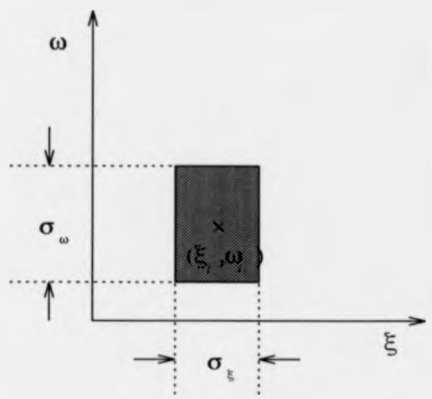


Figure 2.4: Spatial/spatial frequency coefficients within the resolution cell have the resolution limit of the uncertainty principle.

In the spatial domain, the windowing function $w(\xi)$ is centered at ξ_i and has a bandwidth of standard deviation σ_ξ. Its Fourier transform is centered at ω_i and has a bandwidth of σ_ω. A set of coefficients $\hat{x}(\xi, \omega)$ is a description of $x(\xi)$ within the resolution cell of

$$[\xi_i - \sigma_\xi, \xi_i + \sigma_\xi] \times [\omega_i - \sigma_\omega, \omega_i + \sigma_\omega] \tag{2.12}$$

shown in figure 2.4.

A major constraint on the conjoint spatial/spatial frequency representation concerns the maximum degree of simultaneous local energy concentration in both domains, and is usually referred to as the uncertainty principle [40]. This principle states that the product of the bandwidth along each dimension in one domain and the corresponding bandwidth in the other domain cannot be made smaller than some constant, i.e. no signal can be represented arbitrarily compactly in both domains.

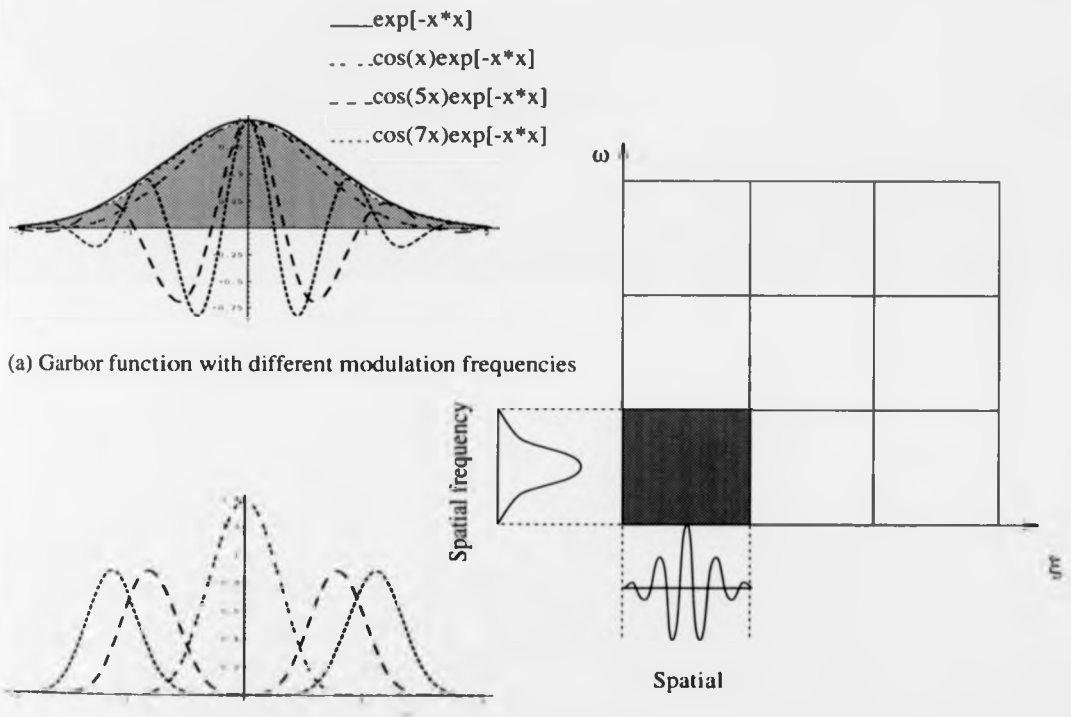
Hence, the resolution cell has to obey the inequality [40] [27]

$$\sigma_\xi \sigma_\omega \geq \frac{\pi}{2} \tag{2.13}$$

Given this constraint, Gabor claimed that the Gaussian windowing function is a suitable filter to provide maximum simultaneous locality in both spatial and spatial frequency domains.

One approach, proposed by Spann and Wilson [104], uses the joint spatial and spatial frequency space in texture analysis. A number of interesting and important features reported in this work has motivated other image analysis work [102]. Jain and Farokhnia [59] used even-symmetric Gabor filters to measure the local energy. Du Buf [32] applied a least-squares approximation to measure the local Gabor energy spectrum in terms of visually relevant texture attributes such as directionality and coarseness. Bovik et al [9] used 2-D Gabor functions as localised filters to encode images into multiple narrow spatial frequency and orientation channels. By analyzing the responses of the filters, it was possible to segregate textural regions of different spatial frequency, orientation or phase characteristics.

The Gabor representation provides a local description in the conjoint spatial/spatial frequency domain, and preserves the symmetry property of translation in the spatial/spatial frequency domain. However, the spatial/spatial frequency resolution cell is of fixed size, as is shown by applying the Fourier transform to equation (2.9) with different frequencies; different cells have the same bandwidth but different spatial frequency position. The limitation of a fixed resolution cell of the Gabor representation leads to the rectangular quantisation of the conjoint spatial/spatial frequency domain shown in figure 2.5. In figure 2.5, (a) depicts the real part of Gabor functions



(a) Garbor function with different modulation frequencies

(b) Fourier transforms of Gabor functions in (a)

(c) Tessellation of spatial/spatial frequency plane by Gabor functions

Figure 2.5: Gabor functions, their Fourier transforms and spatial/spatial frequency plane tessellation by the Gabor function.

of three different frequencies and (b) shows the corresponding Fourier transforms. That the Fourier transforms have the same shape but different positions indicates that the Gabor function has a fixed bandwidth. (c) depicts the spatial/spatial frequency plane tessellation in terms of these elementary Gabor functions; note again that the resolution cell is of fixed size and shape.

The local structure of a texture is hard to characterise by a single scale. An important requirement of such representations is that over the range of possible features the locality in each domain should *not* be fixed. The texture features should have various degrees of locality and extend over different sized areas of the other domain. In order to combine locality at various scales, a multiresolution framework has to be used in the analysis. Multiresolution approaches can be divided into scale-space methods and hierarchical frameworks.

2.5.2 Scale-space Representation

In a scale-space representation, a signal is convolved with Gaussian functions of increasing width. For the signal $f(x)$, with scale σ, spatial co-ordinate ξ, the scale-space representation is defined as

$$F(\sigma, \xi) = \int f(x) w_{\sigma,\xi}(x) dx \tag{2.14}$$

where

$$w_{\sigma,\xi}(x) = \sigma^{\frac{-1}{2}} w(\frac{x - \xi}{\sigma}) \tag{2.15}$$

is the filter kernel. Marr [70] used image features at different resolutions to represent an image. By matching zero-crossing locations of second derivatives through these different resolutions, a single image of boundary estimates may be constructed. Based

on these estimates, the image can be segmented into primitive regions upon which object recognition may be performed. Witkin applied similar ideas to propose a scale-space filtering function [109]. This method is based on smoothing the signal with a filter of continuously increasing scale. The resulting blurred signal is processed by a second derivative operator. This is equivalent to convolution with the second derivative of the Gaussian function, given as

$$\frac{d^2w}{dx^2} = \sigma^{-2}(\frac{x^2}{\sigma^2} - 1)exp[\frac{-x^2}{2\sigma^2}] \tag{2.16}$$

where σ is the standard deviation of the Gaussian function. Koenderink has developed the scale-space representation further into a general approach to vision [64]. Its main weaknesses for texture are that it is computationally expensive and does not provide a natural set of features for analysis.

2.5.3 Pyramid Representation

Normally, a given image and particularly a textured image will have different structure at different resolutions. A pyramid representation of an image uses a hierarchy of regularly spaced resolution levels [79]. Images of different resolution are formed by lowpass filtering, then subsampling to the new Nyquist frequency. At the top level of the pyramid is the most lowpass version of the image, which may be as small as 1×1. At the bottom level of the pyramid is the image of the highest resolution - the original image. A basic pyramid representation can be used in image decomposition and is defined as a function of a logarithmic scale [85], as follows

$$x(\xi_1, \xi_2, l) = \sum_{j=-M}^{M} \sum_{k=-M}^{M} w(j,k)x(2\xi_1 - j, 2\xi_2 - k, l-1) \tag{2.17}$$

where l denotes the scale and $w(\xi_1, \xi_2)$ is a lowpass kernel defining the transformation function between the different scales. The pyramid representation has been applied to different fields of image analysis and particularly coding because most of the image energy is contained in the high levels of the pyramid, which have fewer pixels. Burt [12] has defined a multiresolution representation consisting of regularly spaced resolution levels. He has applied the transform to many aspects of machine vision including texture analysis. Larking and Burt [65] computed local energy spectral estimates upon each of the bandpassed image of the multiresolution bandpass transform, and pixels are classified by energy to perform texture segmentation. Burt [13] suggests the use of edge density instead of spectral density as a texture measure.

Multiresolution pyramid processing is computationally effective, especially for low level image analysis. The computational advantage of performing processing on several resolution levels has been studied in detail by Rabiner et al [24]. The main difference from the windowed Fourier transforms is that in a pyramid, the frequency domain is split into regions having constant *relative* bandwidth. That is, the ratio of bandwidth to centre frequency is fixed, whereas in the windowed Fourier transform the tessellation is in terms of absolute bandwidth. Figure 2.6 shows a pyramid representation where each square in the frequency plane is the region corresponding to one basis vector. The pyramid representation is invertible and deals properly with scale. In order to process structural differences over scale, the pyramid representation [14] utilised the multiresolution framework to extract the difference information between the approximation of an image at two different resolutions [85]. The extraction of the difference is a bandpass filtering function, and Burt and Adelson called this a Laplacian pyramid, because the difference between different resolution levels resem-

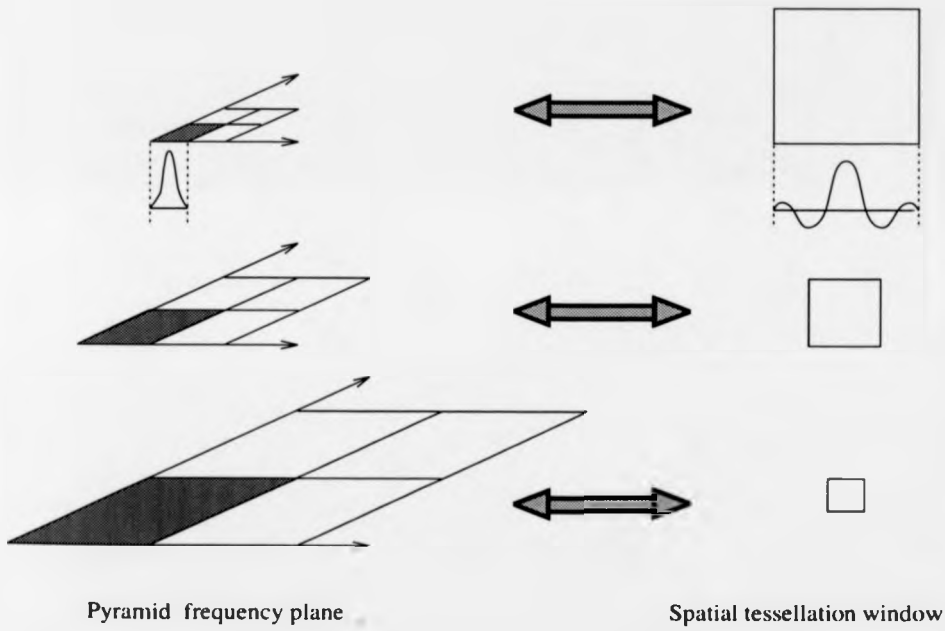


Figure 2.6: Hierarchical windowing allows a pyramid representation to tessellate the spatial/spatial frequency plane over a range of scales.

bles the action of Laplacian-of-Gaussian bandpass filters [14]. The Laplacian of the lowpass Gaussian filter is an isotropic function. In such cases, the transform not only removes the possibility of introducing bias but also reduces the artefacts caused by the rectangular nature of the sampling grid. On the other hand, because it is an isotropic function, the Laplacian-of-Gaussian bandpass filter has restricted utility in estimating orientational features. Moreover, the Laplacian pyramid representation is overcomplete, because the whole representation contains 4/3 times the number of samples in the original image, all of which are needed for the perfect reconstruction.

2.5.4 Wavelet Transform

The wavelet transform is a signal representation which decomposes the signal by expanding the signal into a family of functions having the same bandwidth on a logarithmic scale, obtained by the dilation and translation of a unique function the so called *motherlet*. In the $1-D$ case, the wavelet transform of a signal $f(x)$ is defined as

$$F(\sigma,\xi) = \frac{1}{\sqrt{\sigma}} \int f(x) w(\frac{x-\xi}{\sigma}) dx \tag{2.18}$$

where $w(\xi)$ is the wavelet function, which satisfies some admissibility conditions [49]. The 2-D discrete wavelet transform originates from the Laplacian pyramid. The simplest discrete wavelet transform coefficients are derived by successive decimation and translation operators, using one lowpass filter h and one highpass filter g to retain the same sample count as that of the original image. Since the decimation will introduce aliasing terms, the general trend so far has been to seek an orthogonal set with octave decimation and translation filters. One of these special filtering functions

is a pair of quadrature mirror filters defined in terms of the Z-transform as [100]

$$G(z) = -z^{-1}H(-z^{-1}) \tag{2.19}$$

where G(z) and H(z) are the Z-transforms of the lowpass and highpass filter respectively. In this manner, the aliasing terms caused by the subsampling will be cancelled so that exact reconstruction can be achieved and the redundancy of the representation minimised.

The wavelet frame can be well localised in both time and frequency domains. In phase-space, their time/frequency spread is low and they have good conjoint resolution. Therefore, by representing a signal with such bases, each coefficient corresponds to the magnitude of the *motherlet* at that location (in time or frequency). The discrete wavelet transform is fast to compute, using a tree-based algorithm.

The basic concept of using two filters in the wavelet transform is that the high-pass filter is used to extract 'details' at a given scale, while the low-pass filter is used to smooth the signal to allow the next scale to be explored. As an illustration, figure 2.7 demonstrates the tessellation of the conjoint spatial/spatial frequency domain by the wavelet transform. The spatial co-ordinate ξ and spatial frequency co-ordinate ω in the wavelet transform are related in such a way that cells at high frequencies have low frequency resolution and vice versa. Further reading can be found in Daubechies [26] and Mallat [67].

The conventional wavelet transform appears not to possess a rich enough basis to give a general way of modelling and analysing images. This is because each scale in a 2-D wavelet transform contains only 4 frequency bands [100]. An alternative representation, which contains the wavelet transform and Gabor representation was

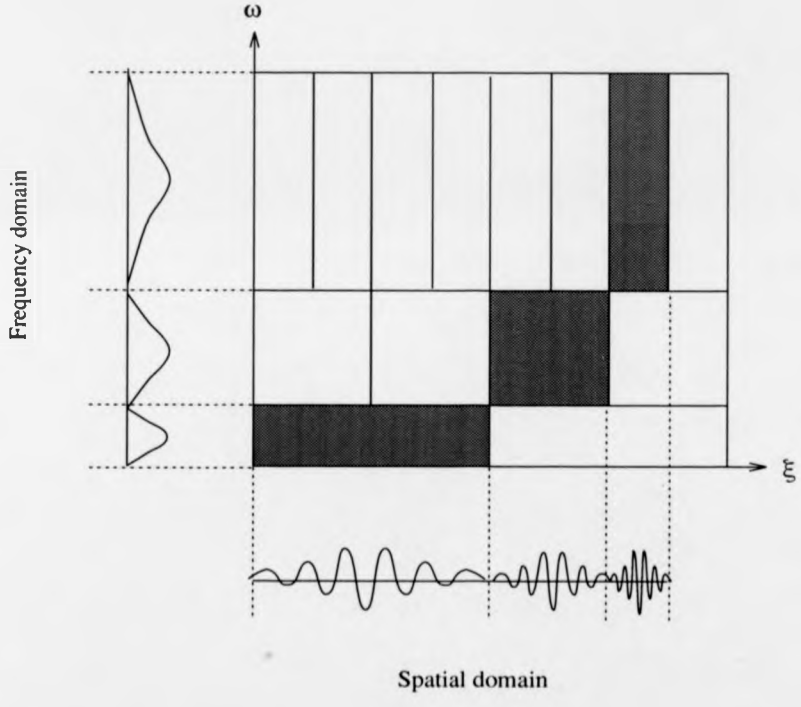


Figure 2.7: WT decomposes the signal into nonuniform spatial/spatial frequency plane cells, high frequencies have high spatial resolution and low frequencies have low spatial resolution.

used by Wilson and Spann [104]. Although the wavelet transform is obviously adequate for dealing with scale-invariant structures such as luminance edges, it seems unlikely that the standard wavelet transform would provide a very effective means of discriminating textures. As reported in [68], the author views the texture element as a particular function of the wavelet orthonormal basis and uses the orientation and narrow frequency tuning to discriminate the textured image. This method has achieved limited success. Both the wavelet transform and the Laplacian pyramid are severely restricted in terms of their representation of orientation. Most researchers in computer vision tend to employ wavelets with some degree of orientation selectivity: the greater the orientation selectivity, the larger the number of wavelets required [104].

2.5.5 Summary

It is well known in signal analysis that the Fourier transform is a linear and invertible transform. The complex exponential functions form a complete orthogonal basis. Symmetries under dilation, translation and rotation are preserved within the transform. Having those sound mathematical characteristics, Fourier analysis appears at first to be a satisfactory candidate as a texture analysis tool. However, the Fourier transform is limited by its nonlocality, as discussed in section 2.5.1.

By employing Gabor functions, the Fourier transform becomes a localised transform. To overcome the fixed resolution of the Gabor signal representation, multiresolution methods have been applied. In the multiresolution pyramid representation, the overcomplete representation restricts its utility in a generalised texture analysis. The Laplacian pyramid is also inadequate because all bands are isotropic. This leads

to consideration of the wavelet transform as a possibly preferable representation.

The wavelet transform and the multiresolution pyramid can both well describe localised image features over different scales. The wavelet transform is an invertible representation which is adapted to some of the natural symmetries of the vision problem. Both methods also have the potential for dealing with the complexity problem by systematic variation of the scale at which the analysis task is performed and provide a rigorous basis for multiresolution analysis. The wavelet transform is computationally fast and requires less storage than the multiresolution pyramid, but the wavelet transform has limitations in the area of symmetry. Because the wavelet transform is based on a 3 parameter subgroup of the 2-D affine transform, it does not include the rotation and shearing that are essential for texture processing. For image representations better suited to texture analysis, a more general transform is required. A combined wavelet-Gabor transform was found effective in [91].

2.6 The Multiresolution Fourier Transform

It was demonstrated above that many of the approaches are not suitable when dealing with analysis problems such as those presented by textural images. This leads to the conclusion that a generalised texture analysis method is needed, in the form of a model that provides characteristics relevant to all perceptually important texture features. To overcome these limitations, a general multiscale Gabor transform, the multiresolution Fourier transform (MFT) [102] is used. Based on this framework, good results have been achieved in low level image analysis [16, 30] and polyphonic audio analysis [77].

The use of the standard Gabor transform implies the use of a fixed size windowing

function. The MFT allows the analysis window size to vary independently of the feature model. The complex exponential filters of the Gabor transform are used effectively as wavelets in a multiscale version of the Gabor transform. This is different from the standard wavelet transform in the choice of the number of mother wavelets used. The MFT is a hybrid combining the features of the wavelet transform and the Gabor transform. It is an extension of the methods used by Spann and Wilson [91].

2.6.1 Definition of the MFT

The MFT is a generalised form of wavelet transform and Gabor transform, which has the ability to represent arbitrary signals of compact support over a range of resolutions in an interference-free manner [102]. In this hierarchical representation form, the MFT can trade off the resolution in spatial and spatial frequency domains, by increasing the resolution in the spatial domain and decreasing the resolution in the frequency domain. In the 2-D case, the top level of the MFT representation is the discrete Fourier transform of the image, and the intermediate levels resemble the WFT with decreasing window size. The bottom level is the original image. The general structure of the MFT is shown in figure 2.8.

Continuous Case

In the $1-D$ continuous case, for a given signal $x(\xi)$, the MFT is

$$\hat{x}(\xi,\omega,\sigma) = \sigma^{1/2}\int_{-\infty}^{\infty} w(\sigma(\chi-\xi))x(\chi)\exp[-j\omega\chi]d\chi \tag{2.20}$$

where σ is the scale factor, ξ is the spatial co-ordinate, ω is the frequency co-ordinate and $w(\xi)$ is an appropriate windowing function. Thus $\hat{x}(\xi,\omega,\sigma)$ is a windowed Fourier spectrum of a local region in the signal domain centered at spatial position ξ at scale

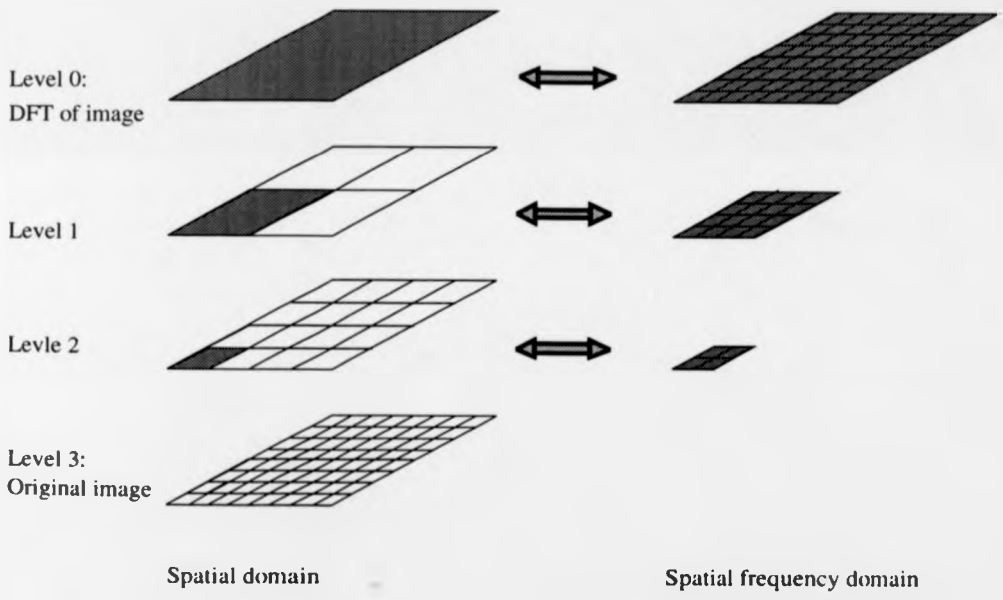


Figure 2.8: 2-D structure MFT.

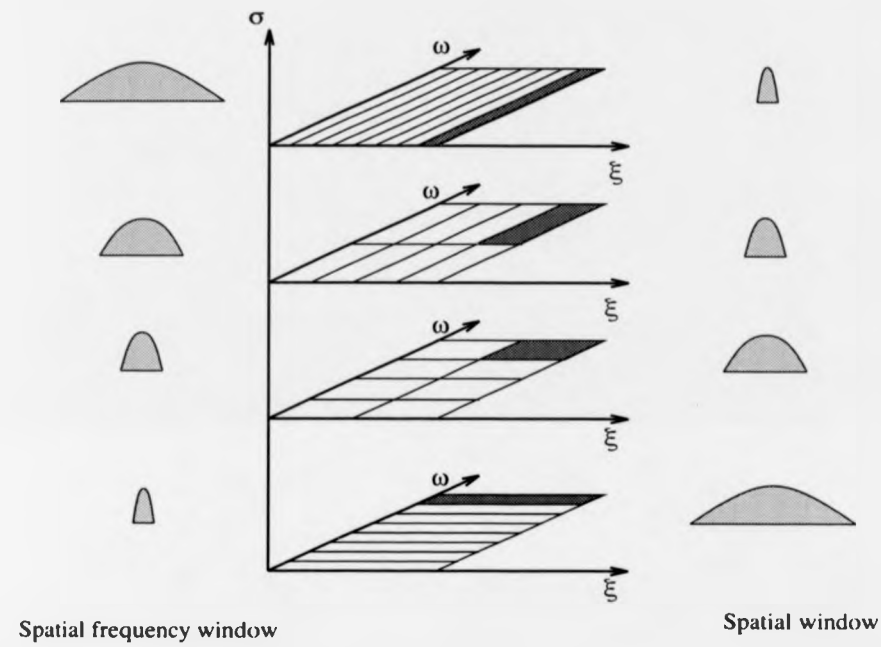


Figure 2.9: Tessellation of Multiresolution Fourier transform.

σ with windowing function $w(\xi)$. The MFT tessellation adopts a generalised scale-space windowed Fourier transform, (see figure 2.9). The extension to m-D continuous functions is straightforward. Let $\vec{\xi} = (\xi_1, ..., \xi_m)^T$, then

$$\hat{x}(\vec{\xi}, \vec{\omega}, \sigma) = \sigma^{m/2} \int_{-\infty}^{\infty} w(\sigma(\vec{\chi} - \vec{\xi}))x(\vec{\chi}) \exp[-j\vec{\omega} \cdot \vec{\chi}]d\vec{\chi} \qquad (2.21)$$

where $\vec{\omega}$ is the Fourier co-ordinate.

Discrete Case

The 1-D MFT coefficient at scale $\sigma(n)$, position $\xi_i(n)$ and frequency $\omega_j(n)$ can be written as

$$\hat{x}(\xi_i(n), \omega_j(n), \sigma(n)) = \sum_k w_n(\xi_i(n) - \xi_k) x(\xi_k) \exp\left[-\jmath \xi_k \omega_j(n)\right] \tag{2.22}$$

where $w_n(\xi_k)$ is the scale n window sequence, ξ_k is the kth sample point for the original sequence and the sample points at scale n are $\xi_k(n)$ and $\omega_l(n)$ for the original and Fourier co-ordinates respectively. Extension of the 1-D case in (2.22) to the 2-D case can be achieved using a cartesian separable implementation.

2.6.2 MFT Properties

The requirements of a general signal representation are retained in the MFT in the following respects [103]:

1. Linearity - the WFT function is linear by definition, thus the MFT shares this property.

2. Invertibility - since the WFT function is invertible, this property is preserved in the MFT.

3. Overcompleteness - because the MFT is a stack of WFT's, it contains more samples than the original signal.

4. Symmetry - to illustrate the symmetry properties of the MFT, consider the kernel $g_{\vec{\xi},\vec{\omega},\sigma}(\vec{\chi})$ in the MFT

$$g_{\vec{\xi},\vec{\omega},\sigma}(\vec{\chi}) = w(\sigma(\vec{\chi} - \vec{\xi})) \exp[-\jmath \vec{\omega} \cdot \vec{\chi}] \tag{2.23}$$

Translation of such a function by $\vec{\delta}$ gives the function

$$w(\sigma(\vec{\chi}+\vec{\delta}-\vec{\xi}))\exp[-\jmath\omega\cdot(\vec{\chi}+\vec{\delta})] = \exp[-\jmath\vec{\omega}\cdot\vec{\delta}]g_{\vec{\xi}-\vec{\delta},\vec{\omega},\sigma}(\vec{\chi}) \tag{2.24}$$

while a shift in frequency by $\vec{\rho}$ gives

$$w(\sigma(\vec{\chi}-\vec{\xi}))\exp[-\jmath(\vec{\omega}+\vec{\rho})\cdot\vec{\chi}] = g_{\vec{\xi},\vec{\omega}+\vec{\rho},\sigma}(\vec{\chi}) \tag{2.25}$$

and a dilatation by a factor ϵ gives

$$w(\sigma(\epsilon\vec{\chi}-\vec{\xi}))\exp[-\jmath\vec{\omega}\cdot\epsilon\vec{\chi}] = g_{\epsilon^{-1}\vec{\xi},\epsilon\vec{\omega},\epsilon\sigma}(\chi) \tag{2.26}$$

and a rotation by an angle θ gives, for an isotropic window function,

$$w(\sigma(\vec{\chi}-\vec{\xi}))\exp[-\jmath\vec{\omega}_\theta\cdot\vec{\chi}] = g_{\vec{\xi},\vec{\omega}_\theta,\sigma}(\vec{\chi}) \tag{2.27}$$

where $\vec{\omega}_\theta$ is the rotation of $\vec{\omega}$ by an angle θ. The vectors thus transform amongst themselves under the action of a group of symmetries which is central to image analysis [27]. These symmetry properties are a key component in the texture model discussed in Chapter 4.

5. Locality - the MFT uses scale σ as part of a signal representation in spatial/spatial frequency domains. By choosing a coefficient set within a localised resolution cell, the analysis task can be performed locally.

2.7 Concluding Remarks

A generalised tool for texture analysis has been set forth in the present chapter. The MFT seems to have the right combination of symmetry properties and locality to overcome the limitations of earlier Fourier domain approaches to texture analysis. It has been shown to be effective in a range of image analysis problems [16] [30].

Chapter 3

The MFT implementation

3.1 Introduction

In association with the MFT definition given in Chapter 2, it is convenient in this chapter to consider Fourier analysis in the context of a bank of filters. This leads to both theoretical and practical insights into the MFT. The best window function is chosen on the basis of the uncertainty principle and a scheme for its generation is reviewed. Two different MFT implementations are discussed. Some examples of MFT outputs are presented. The chapter establishes the conclusion that the MFT provides a general signal analysis representation which is efficiently implemented and suitable for texture analysis. It should not cause confusion that the first index of the notation adopted in this chapter is used as the spatial domain co-ordinate and the second is the frequency domain co-ordinate.

3.2 Interpretations of the MFT

The MFT is a transform that converts signals $x(\xi_i)$ of one-dimensional support into a representation $\hat{x}(\xi_i, \omega_k, \sigma)$ of three-dimensional support. As described in Chapter 2, it consists of a stack of windowed Fourier transforms over a range of scales of the

window, so that for a given scale σ, the MFT becomes a windowed Fourier transform. As noted in Chapter 2, the WFT is a spatial/spatial frequency representation with two-dimensional support (ξ_i, ω_k), where ξ_i is the spatial variable and ω_k the spatial frequency variable. By fixing one of these two variables, the MFT can be interpreted either as a conventional Fourier transform or as a linear filter bank scheme. It follows that there are two corresponding implementations.

3.2.1 Fourier Transform Interpretation

Since the MFT is a family of WFTs, the MFT for a given level can be defined as

$$\hat{x}(\xi_i, \omega_k) = \sum_{\xi'=0}^{M-1} x(\xi') w(\xi_i - \xi') \exp[-\jmath \xi' \omega_k] \tag{3.1}$$

where $w(\xi_i - \xi')$ is a window sequence, M is the size of $x(\xi)$. Equation (3.1) can be viewed as measuring the Fourier transform of the signal $x(\xi')$ at spatial position ξ_i, seen through a window with response $w(\xi')$, as shown in figure 3.1. The signal has a fixed origin and as ξ_i changes, the window slides along the signal domain so that at each value of ξ_i, a different portion of the signal is viewed. That is, $\hat{x}(\xi_i, \omega_k)$ is the Fourier transform of the sequence $x(\xi')w(\xi_i - \xi')$, and can be thought of as a function of ω_k for fixed ξ_i. Because multiplication by the relatively short window $w(\xi_i - \xi')$ effectively suppresses the signal outside a neighbourhood around the value $\xi' = \xi_i$, the coefficient $\hat{x}(\xi_i, \omega_k)$ is simply a local spectrum of the signal $x(\xi')$ around the value ξ_i.

The fact that for a given value of ξ_i, $\hat{x}(\xi_i, \omega_k)$ has the same properties as a conventional Fourier transform leads to the result that $x(\xi')$ can be recovered exactly from the transformed coefficients if the window $w(\xi_i - \xi')$ has at least one nonzero element

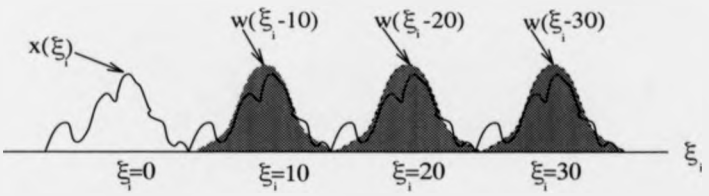


Figure 3.1: Representation of the MFT at a given level can be interpreted as a conventional Fourier transform.

[3]. Specifically, from the Fourier transform

$$x(\xi')w(\xi_i - \xi') = \frac{1}{M}\sum_{\omega_k=0}^{M-1} \hat{x}(\xi_i, \omega_k)\exp[j\xi'\omega_k], \tag{3.2}$$

from which it follows that

$$x(\xi') = \frac{1}{Mw(0)}\sum_{\omega_k=0}^{M-1} \hat{x}(\xi_i, \omega_k)\exp[j\xi'\omega_k] \tag{3.3}$$

if $w(0) \neq 0$.

3.2.2 Filter Bank Interpretation

A rearrangement of the order in equation (3.1) leads to another useful interpretation of the MFT at a given level. If we exchange the order of $x(\xi')$ and $w(\xi_i - \xi')$ in equation (3.1), then $\hat{x}(\xi_i, \omega_k)$ can be rewritten as

$$\hat{x}(\xi_i, \omega_k) = \sum_{\xi'} w(\xi_i - \xi')x(\xi')\exp[-j\xi'\omega_k] \tag{3.4}$$

It is obvious from equation (3.4) that for each value of ω_k, $\hat{x}(\xi_i, \omega_k)$ is the convolution of the window function $w(\xi_i - \xi')$ with the modulated signal $x(\xi')\exp[-j\xi'\omega_k]$. Thus, for a particular value of ω_k, $\hat{x}(\xi_i, \omega_k)$ can be thought of as the output of the linear filter system depicted in figure 3.2.

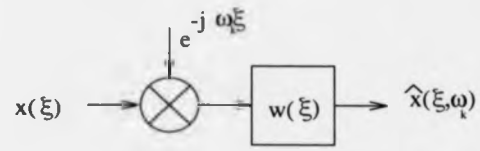


Figure 3.2: The MFT at a given level can be interpreted as a lowpass filter.

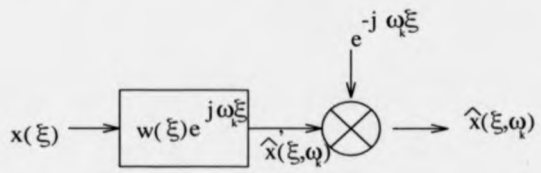


Figure 3.3: The MFT at a given level can be interpreted as a bandpass filter.

Another interpretation of $\hat{x}(\xi_i, \omega_k)$ in terms of linear filtering is evident from the following equation

$$\begin{aligned} \hat{x}(\xi_i, \omega_k) &= \sum_{\xi'} x(\xi_i - \xi') w(\xi') \exp\left[-\jmath(\xi_i - \xi')\omega_k\right] \\ &= \exp\left[-\jmath\omega_k \xi_i\right] \sum_{\xi'} x(\xi_i - \xi') w(\xi') \exp\left[\jmath\omega_k \xi'\right] \end{aligned} \tag{3.5}$$

As shown in figure 3.3 $\hat{x}(\xi_i, \omega_k)$ can also be thought of as the result of modulating $\exp[-\jmath\omega_k\xi]$ with the output of a complex bandpass filter whose impulse response is $w(\xi')\exp[\jmath\omega_k\xi']$.

If the MFT is sub-sampled by a factor of S, reconstruction is still possible provided that the window functions and sampling intervals are chosen correctly. The reconstruction of the signal $x(\xi)$ of size M can be obtained in terms of the linear filter system by modulating $\hat{x}(\xi_i, \omega_k)$ back to frequency ω_k, and summing the result

over S frequency channels.

$$x(\xi) = \sum_{i} \sum_{k=0}^{S-1} f(\xi - \xi_i)\hat{x}(\xi_i, \omega_k) \exp[j\omega_k \xi] \tag{3.6}$$

where $f(\xi)$ is the synthesis window.

Figure 3.4 shows a simplified block diagram of a filter bank. The signal is considered as a sequence $x(\xi)$; the analysis filters $w(\xi)$ are convolved with the modulated signal; down-sampling by a factor of S, leads to the coefficients $\hat{x}(\xi, \omega)$, which form the representation of $x(\xi)$. For reconstruction, the coefficient set $\hat{x}(\xi, \omega)$ are interpolated with the synthesis filter $f(\xi)$ and modulated with $\exp[j\omega_k \xi_i]$ leading to the synthesised signals $x_k(\xi)$. These are summed to retrieve the original signal $x(\xi)$ [81]. In this system, the input signal $x(\xi)$ is passed through S filters. The bank of filters separates the frequency spectrum of interest into disjoint equal-width bands. The spacing of the filters is arranged such that they cover the frequency spectrum. Two important considerations go into the implementation of filter bank systems. One of the issues is the sampling rate to ensure the transform is invertible. The other which has to be resolved is the type of filters to be used, i.e. the window functions.

3.3 Sampling Scheme

A level of the MFT transform may consist of separate Fourier representations of contiguous spatial blocks. For a finite sequence $x(\xi)$ of length $M = 2^n$, the MFT representation consists of $n + 1$ levels, each having M samples. A given level l of the MFT is divided into 2^l frequency domain sets, each containing 2^{n-l} spatial samples. These sets represent disjoint bands covering the frequency domain. Each frequency set is produced by convolving modulated $x(\xi)$ with the analysis window $w(\xi)$, and

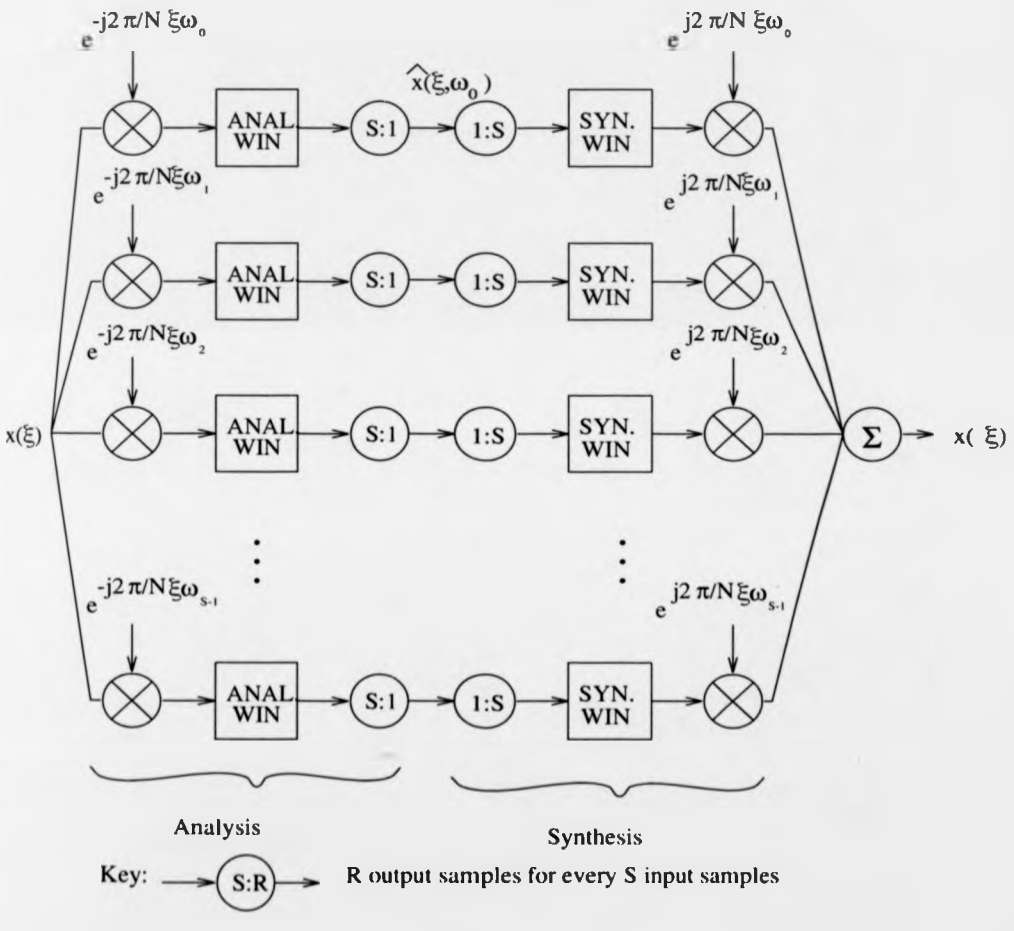


Figure 3.4: The MFT combines analysis filters and synthesis filters corresponding to forward transforms and inverse transforms respectively.

subsampling by a factor equal to 2^l.

The MFT is a spatial/spatial frequency representation, so that the sampling is done in both the frequency and spatial domains. For a given level l, the number of samples within a block in the spatial domain is S, while in the frequency domain it is N. From the sampling theorem, a signal reconstruction can be obtained only if the number of sampled coefficients is at least equal to that of the original signal. That is,

$$S\,N \geq M \tag{3.7}$$

It is termed as a *critical sampling* in this work if the number of coefficient samples is equal to the number of samples of the original signal [27].

It can be shown that critical sampling will introduce bias along the edge of the window because of the characteristics of the finite Fourier transform. One solution is to increase the number of samples to remove the edge effect. The number of coefficients chosen in this work is two times that of the original signal [16],

$$S\,N = 2M \tag{3.8}$$

which results in each level of the MFT representation being of size $2M$. In order to achieve this oversampling, the window function has to overlap 50% for each adjacent sampling interval as shown in figure 3.5. Note that this redundancy is different from that of the pyramid representation discussed in Chapter 2.

As discussed in Chapter 2, each level of the MFT is a complete representation. In order to satisfy this requirement, on successive levels of a 1-D MFT, the number of frequency samples increases by a factor of two, and the number of spatial samples within each set decreases by the same factor.

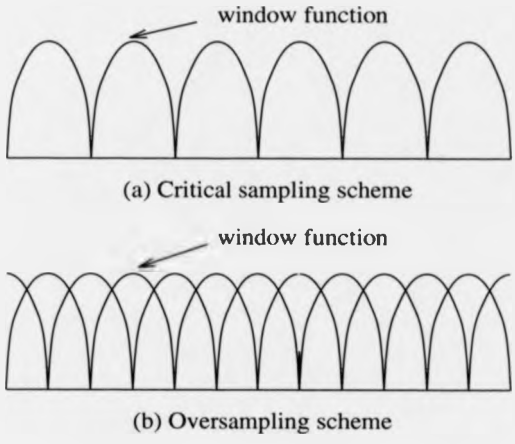


Figure 3.5: The analysis filters are disjoint for critical sampling scheme and overlapping 50% for the oversampling scheme.

3.4 Choice of Window Function for the MFT

The primary purpose of the window in the windowed Fourier transform is to limit the extent of the input signal to be transformed so that the spectral characteristics are reasonably 'stationary' over the duration of the window. The more rapidly the signal characteristics change, the shorter the window should be. That is, as the window becomes shorter, frequency resolution decreases, while the ability to resolve spatial changes increases. By contrast, the use of band-limited window functions in the frequency domain reduces spatial resolution. According to the uncertainty principle [40], a signal can not be both index-limited and band-limited. Consequently, the choice of window length and other properties involves a trade-off between frequency resolution and spatial resolution.

The resolution trade-off can be made by a window with finite bandwidth but which

is concentrated as much as possible in the time domain. The window function $w(\xi)$ used in the MFT is a band-limited Finite Prolate Spheroidal Sequence (FPSS). The FPSS is used as a windowing function with maximal simultaneous energy concentration in both spatial and frequency domains [16].

3.4.1 Finite Prolate Spheroidal Sequences

The importance of the Prolate Spheroidal Sequences (PSS) introduced by Slepian *et al* [89] in signal processing has long been well established. The property of the FPSS of simultaneous energy concentration in both the spatial domain and the corresponding frequency domain is clearly related to the uncertainty principle as observed by Gabor [40]. As a result, the sequences are widely used in signal analysis for filter design [94, 76], data compression [107] and spectrum analysis [93]. In the field of image processing, 2-D versions of the sequences have received considerable attention and many researchers have indicated that the uncertainty principle imposes a limitation on processing operations [75] [28] [108].

The band-limited form of FPSS can be expressed as vector $\boldsymbol{v}$, which is the solution of an eigenvalue problem generated in accordance with constraints in both spatial and frequency domains. The 1-D FPSS problem can be expressed as an eigenvalue problem in the following form [106]. Let T_{n_1,n_2} represent the truncation operator defined as

$$T_{n_1,n_2} = \begin{cases} \delta_{ij} & n_1 \leq i < n_2 \\ 0 & \text{else} \end{cases} \tag{3.9}$$

and B_{k_1,k_2} be the band-limiting operator,

$$B_{k_1,k_2} = F^* T_{k_1,k_2} F \tag{3.10}$$

where F is the DFT matrix. Note that T, B and F are all square matrices. The eigenvalue equation for the band-limited FPSS is then

$$BT\boldsymbol{v}_k = \lambda_k \boldsymbol{v}_k \tag{3.11}$$

where $0 \leq \lambda_k \leq 1$ is the kth eigenvalue and the $\boldsymbol{v}_k$ are the FPSS's. The eigenvalue represents the energy remaining in the eigenvector $\boldsymbol{v}_k$ after the operator BT is applied to it [106]. Thus, the smallest energy loss FPSS corresponds to the largest eigenvalue λ_0.

3.4.2 Solution of the FPSS problem

General methods for the solution of eigenvalue problems are readily available [106]. However the FPSS problem possesses properties which enable its solution to be obtained more easily by successive approximation than the computationally expensive general method.

In the FPSS problem, only the eigenvector corresponding to the largest eigenvalue is required due to the minimum uncertainty property [89]. To this end, a successive approximation method is helpful particularly when the size of the required FPSS, index-limiting operator and band-limiting operator are all powers of two. The method uses the principle that a sequence band-limited (or index-limited) to a given range can be processed to generate solutions for vectors of any size greater than that range. The limiting in the other domain can be adjusted proportionally in respect of the adjusted order. A FPSS problem solved at an adjusted order and adjusted limiting range can then approximate a solution for an original order and limiting range, since the solution sequence will be limited to the fixed range chosen. Successive approximation using

the FPSS problem operator BT can be used to force the approximation to converge to the true value.

The above principle can be used to reduce a given order FPSS problem to a lower order problem, whose solution is less computationally expensive and after interpolation will provide a reasonable approximation to the required solution. Successive application of the BT operator and normalisation will then yield the solution.

The implementation of the scheme for generating FPSS is based on the successive approximation process described above and is fully detailed in [106]. It can be formulated as

step1: Reduce the size of the FPSS by a factor of 2^k, so that the length is small enough for rapid computation.

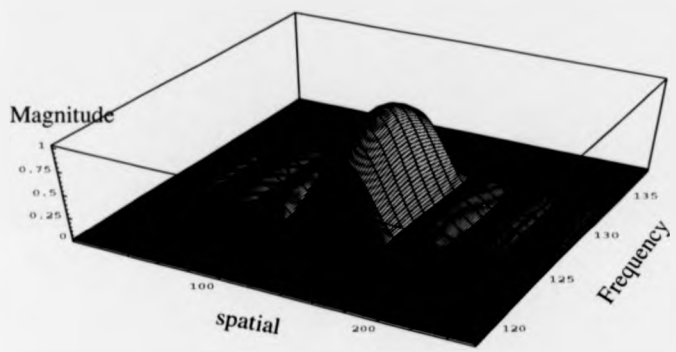
step2: Calculate the smaller FPSS by the general eigenvalue method.

step3: Successively interpolate the reduced size sequence generated in step 2 by factors of 2 and apply the BT operator to obtain an approximation of the required sequence; repeat iterations until minimum error ($BT\boldsymbol{v} - \lambda\boldsymbol{v}$) is reached, normalising the sequence to unit energy at each step.

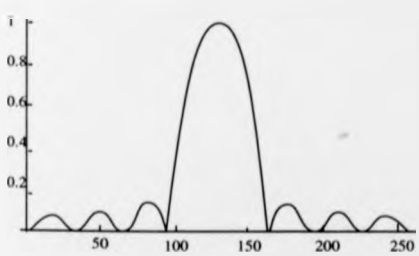
The FPSS used in the oversampled MFT is obtained by increasing the bandwidth by a factor of 2. The resulting sequence $\boldsymbol{u}_l$ is related to the original sequence $\boldsymbol{v}$ by

$$\boldsymbol{u}_l = B_{2N_f} T \boldsymbol{v} \tag{3.12}$$

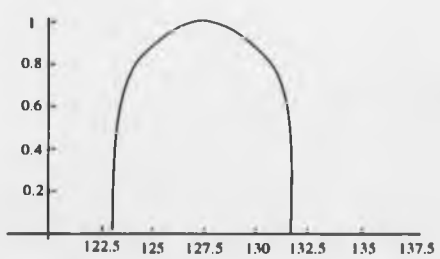
where N_f is the size of the original band-limiting operator. Both the normal and oversampled FPSS's are shown in figure 3.6 and 3.7, (a) is the spatial/spatial frequency domain plot, (b) and (c) show the cross sections in the spatial domain and



(a) 3-D plot of the normal FPSS in spatial/spatial frequency domain

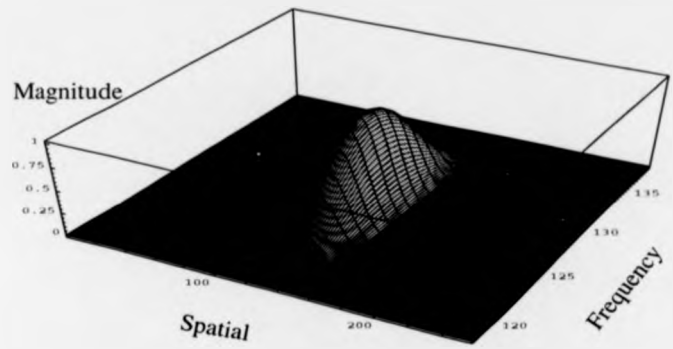


(b) FPSS magnitude in spatial domain

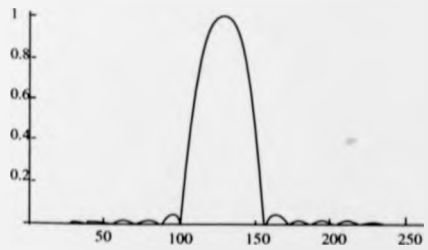


(c) Frequency response of FPSS

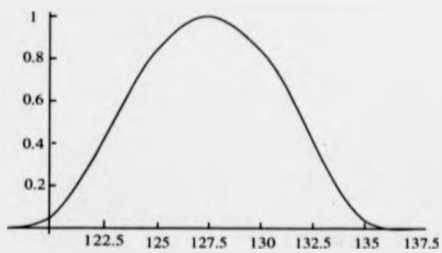
Figure 3.6: The normal windowing FPSS in the MFT.



(a) 3-D plot of the relaxed FPSS in spatial/spatial frequency domain



(b) FPSS magnitude in spatial domain



(c) Frequency response of FPSS

Figure 3.7: The oversampled FPSS window.

spatial frequency domain respectively. From these figures, the largest sidelobe magnitude in the normal FPSS is obviously larger than that in the oversampled FPSS. Thus, interference from parts of the signal outside the main window is reduced in the oversampled case.

3.5 Implementation

Implementation of the MFT can be carried out in two different ways, due to the duality of the Fourier transform. First is the spatial domain implementation using spatial windowing, whereas the other is the spatial frequency domain implementation using a band-limiting window function.

3.5.1 Spatial Implementation

Critical Sampling

Consider initially only the generation of one level of the 1-D MFT for an input signal size of M. If the input signal is partitioned into N blocks of S samples

$$S = \frac{M}{N} \tag{3.13}$$

then the local spectrum can be found as (cf. (3.1))

$$\begin{aligned} \hat{x}(n,k) &= \sum_{m=0}^{M-1} x(m)w(nS-m)e^{-j\frac{2\pi}{M}mkN} \\ &= \sum_{m=0}^{M-1} x(m)w(nS-m)e^{-j\frac{2\pi}{S}mk} \quad 0 \le n < \frac{M}{S} \end{aligned} \tag{3.14}$$

For $n = n_0$, the MFT coefficient $\hat{x}(n_0,k)$ is an estimate of the Fourier spectrum of a local region in the signal domain centred at $n_0S + \frac{S}{2}$. The region corresponds to weighting the signal by a window, which is an index-limiting function so that equation

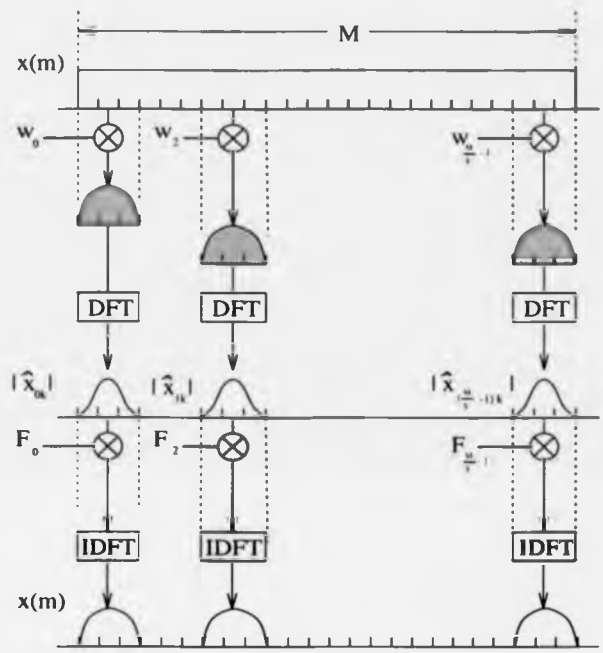


Figure 3.8: The spatial implementation of the MFT.

(3.14) can be written as

$$\hat{x}(n_0, k) = \sum_{m=n_0 S}^{(n_0+1)S-1} x(m) w(n_0 S - m) e^{-j\frac{2\pi}{S}mk} \tag{3.15}$$

which is a simple Fourier transform of the local signal.

The outline of the MFT implementation is schematically depicted in figure 3.8 where an input sequence $x(m)$, of M samples, is partitioned into blocks of S samples by weighting with the window functions $w_r(n)$. Each S-sample block is transformed to the frequency domain via the FFT to generate the MFT local spectrum. The inverse procedure is illustrated at the bottom of the figure.

The 2-D implementation can be accomplished straightforwardly by employing the

cartesian separable product of its 1-D counterparts.

Oversampling

In order to increase frequency localisation, the normal spatial MFT equation (3.14) is modified as

$$\hat{x}(n_0,k) = \sum_{m=0}^{S-1} x(m)w(n_0\frac{S}{2} - m)e^{-j\frac{2\pi}{S}mk} \tag{3.16}$$

By increasing the truncation width in the spatial domain by a factor of two as in equation (3.16), there are twice as many MFT coefficients as there were samples of the original 1-D signal. For the oversampling scheme, the implementation is accomplished by overlapping the window functions 50% as discussed in Section 3.3.

The inversion can be accomplished by two possible procedures. One is to find the synthesis inverse window function $f(m)$ such that

$$w(m)f(m) = \begin{cases} 1 & \text{if } \frac{S}{4} \leq m < \frac{3S}{4} \\ 0 & \text{otherwise} \end{cases} \tag{3.17}$$

In this case, the original signal is reconstructed exactly. Figure 3.9 shows this reconstruction.

The other possible procedure to reconstruct the original signal makes use of the overlapping of the windowing functions. It is sufficient to require that the summation of the analysis-synthesis windows produces a 'flat' overall response. It has been noted that such a choice of reconstruction results in an inversion without error [16]. This suggests that a reconstruction can be found based on the adoption of cosine based windows, so that the product of the analysis and synthesis windows which satisfies

$$w'(m)f'(m) = \begin{cases} \cos^2(\frac{\pi(m-\frac{S}{2})}{2S}) & \text{if } m \leq S \\ 0 & \text{otherwise} \end{cases} \tag{3.18}$$

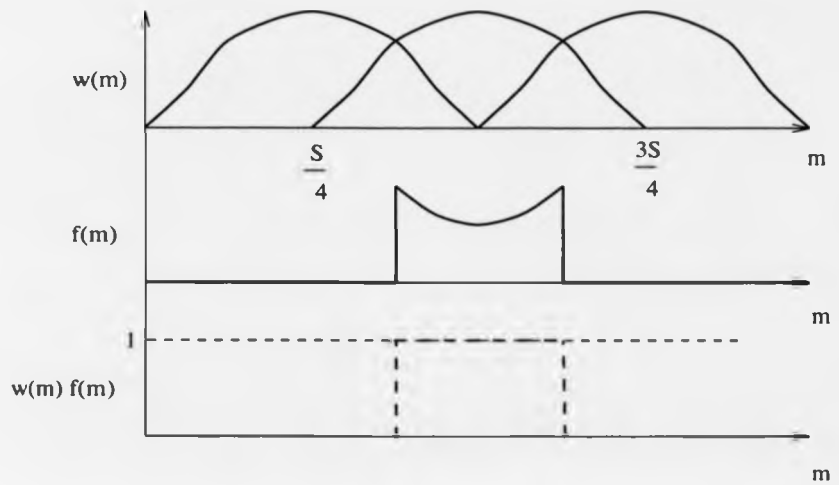


Figure 3.9: Exact reconstruction of the MFT using (3.17).

This method has been employed both in the spatial implementation and in the frequency implementation of the MFT reconstruction (cf. section 3.5.2).

3.5.2 Frequency Implementation

Critical Sampling

Some useful notations are defined before starting the development. Consider a given sequence $x(n)$ of size M and its M point DFT

$$\hat{x}(k) = \sum_{n=0}^{M-1} x(n) e^{-j\frac{2\pi}{M}nk} \tag{3.19}$$

and a window function $w(n)$, with DFT

$$\hat{w}(k) = \sum_{n=0}^{M-1} w(n) e^{-j\frac{2\pi}{M}nk} \tag{3.20}$$

It is the simplest to consider the 1-D signal of size M. The MFT coefficient $\hat{x}(n,k)$ at a given frequency $k = k_0$ is obtained by the convolution relationship (cf.(3.4)).

$$\hat{x}(n,k_0) = \sum_{m=0}^{M-1} x(m) w(nS-m) e^{-j\frac{2\pi}{S}mk_0} \tag{3.21}$$

where S divides M. Some substitutions of the variables in equation (3.21) gives

$$\begin{aligned}
\hat{x}(n,k_0) &= \sum_{m=0}^{M-1} x(m) \sum_{k=0}^{M-1} \hat{w}(k) e^{j\frac{2\pi}{M}(nS-m)k} e^{-j\frac{2\pi}{M}m(Nk_0)} \\
&= \sum_{k=0}^{M-1} \hat{w}(k) \left[\sum_{m=0}^{M-1} x(m) e^{-j\frac{2\pi}{M}m(Nk_0+k)} \right] e^{+j\frac{2\pi}{M}(nS)k} \\
&= \sum_{k=0}^{M-1} \hat{w}(k) \hat{x}(Nk_0+k) e^{+j\frac{2\pi}{M}(nS)k} \\
&= \sum_{k=0}^{M-1} \hat{w}(k) \hat{x}(Nk_0+k) e^{+j\frac{2\pi}{N}nk}
\end{aligned} \tag{3.22}$$

In this work, $w(k)$ is a band-limited function satisfying

$$\begin{aligned}
\hat{w}(k) &\neq 0 \quad 0 \leq k \leq N-1 \\
&= 0 \quad \text{else}
\end{aligned} \tag{3.23}$$

Therefore, equation (3.22) can be written as

$$\hat{x}(n,k_0) = \sum_{k=0}^{N-1} \hat{w}(k) \hat{x}(Nk_0+k) e^{j\frac{2\pi nk}{N}} \tag{3.24}$$

This is simply the IDFT of a multiplication of the spectrum of the analysis window and the appropriately shifted signal spectrum, suggesting that the MFT can be efficiently implemented in the frequency domain, making use of the FFT.

Hence, the procedure for computing the MFT coefficients $\hat{x}(n,k)$ is the following. The M-point DFT of the input sequence $x(n)$ of size M is taken (and only taken once). The shifted input spectrum is then multiplied by the band-limited window

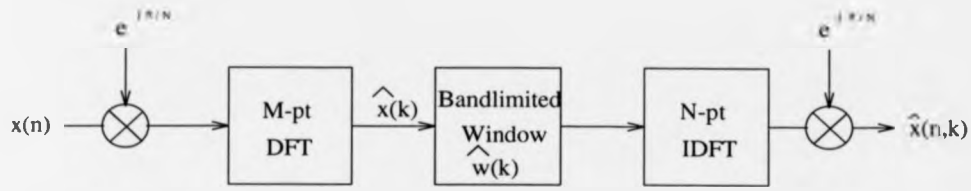


Figure 3.10: The forward MFT implementation. (Note: $M > N$)

$\tilde{w}(k)$ of size N. Thus, the resulting weighted sequence is partitioned into sections each of length N , the IDFTs of which are computed by means of the FFT algorithm to give the desired coefficients $\tilde{x}(n,k)$.

The outline of the forward MFT transform is depicted in figure 3.10. In this figure, there is a complex exponential multiplication to the signal $x(n)$ before applying the DFT. Multiplication by a complex exponential in one domain is equivalent to a shift in the other domain, according to the property of the DFT [10]. Hence, the effect of the multiplication by $e^{j\frac{\pi}{N}}$ is to shift the spectrum spatially by half of one sample so that the frequency samples are placed symmetrically about $\omega = 0$, with $N/2$ samples on each side of $\omega = 0$. Figure 3.11 illustrates this procedure. Figure 3.11(a) depicts the result of an N-DFT of an input signal $x(n)$ where N is a power of 2. Figure 3.11(b) depicts the shift of the spectrum after the phase correction.

Reconstruction of the original signal can be accomplished by a process similar to its generation, by taking the DFT of each row of the MFT coefficients and then applying the analysis vector and deriving the inverse DFT of the result. However, the current implementation is based on the cosine-squared function to give an overlapped

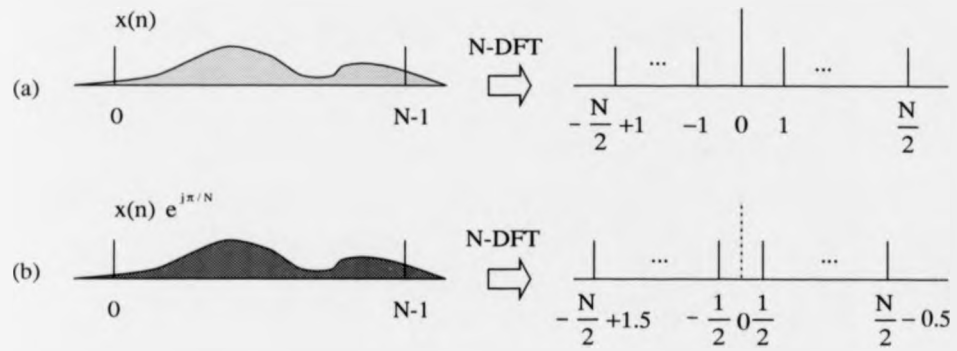


Figure 3.11: The effect of a half sample shift (Note that different shading means phase difference).

reconstruction.

The 2-D MFT is implemented in a cartesian separable manner. That is, the 2-D analysis vector is simply the Kronecker product of its 1-D counterparts. As a 2-D MFT example, consider a given image spectrum $\hat{x}(k)$ of size $M = 32 \times 32$ truncated by a band-limited window function $\hat{w}(k)$ of size $N = 4 \times 4$; then the corresponding spatial sampling grid is $S = 8 \times 8$. The sampling of the convolved result is illustrated by figure 3.12, which shows a frequency spectrum $\hat{x}(k)$ obtained by applying the 32 point 2-D Fourier transform on $x(n)$; the window function separates the image spectrum into 64 4×4 blocks; taking the 4×4 inverse Fourier transform of each block and rearranging the order gives the desired local spectra.

Notice that a bigger band-limiting window size N gives better spatial resolution, whereas the frequency resolution is reduced. For the extreme case, if $N = M$ then the result of the MFT is the original image, and when $S = M$ (or $N = 1$) then it results in the DFT of the input image.

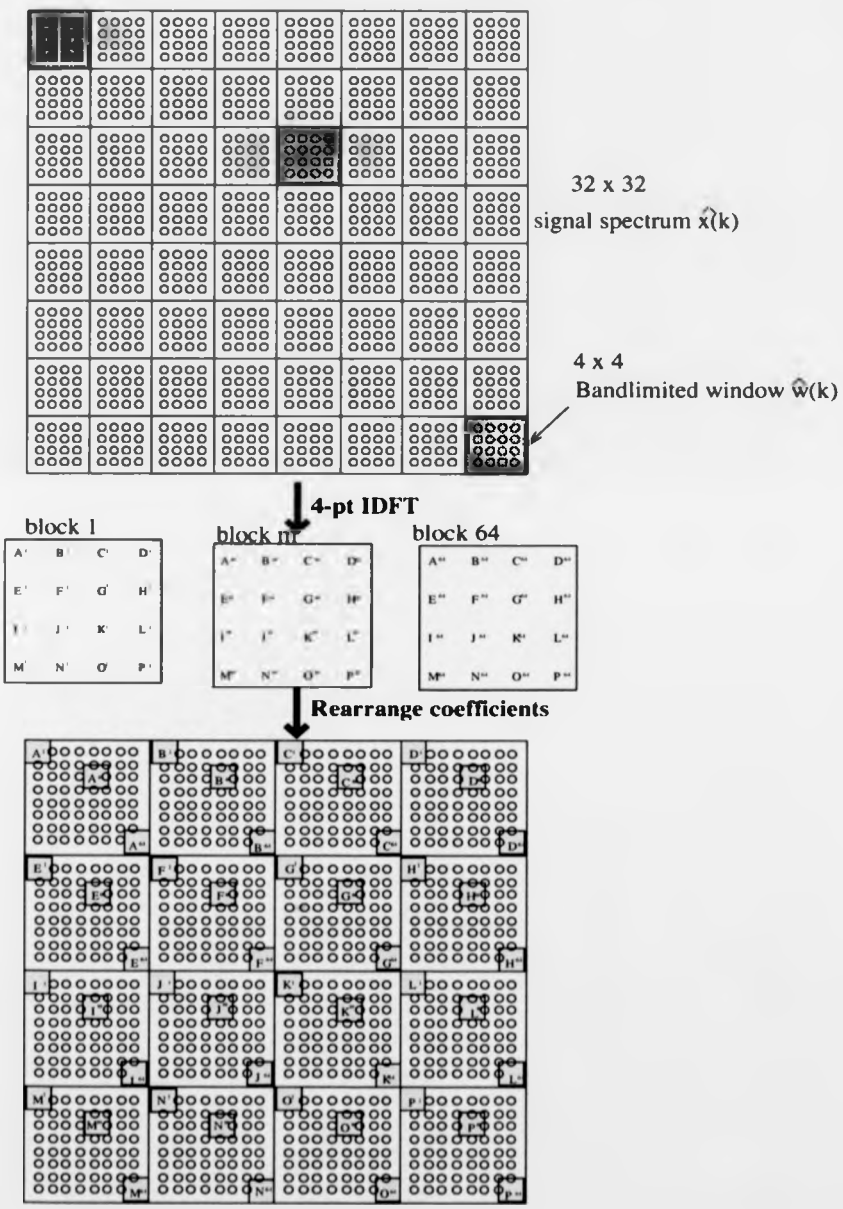


Figure 3.12: The 2-D MFT is generated from convolution of analytic window spectrum with frequency shifted version of input signal.

Oversampling

By analogy with the oversampling in the spatial implementation, the band-limited analysis functions are overlapped by 50% for every adjacent interval, which results in a total of $4M$ MFT coefficients on each level of the 2-D transform, i.e. at each window scale.

3.6 Preliminary Results

This section describes a set of experiments designed to demonstrate the potential of the MFT. In order to illustrate the representation of texture, some textured images are generated.

3.6.1 Synthetic Textured Images

Consider a set of randomly distributed impulses $x(m,n)$ defined in relation to a random variable $u(m,n)$ drawn from a uniform distribution $U(0,1)$:

$$x(m,n) = \begin{cases} 1 & u(m,n) \leq T \\ 0 & u(m,n) > T \end{cases} \tag{3.25}$$

where $u(m,n) \in U(0,1)$, $0 \leq T \leq 1$. A texture field $y(m,n)$ is synthesised by convolving such a set of random distributed impulses $x(m,n)$ with a spatial filter with impulse response $g(m,n)$:

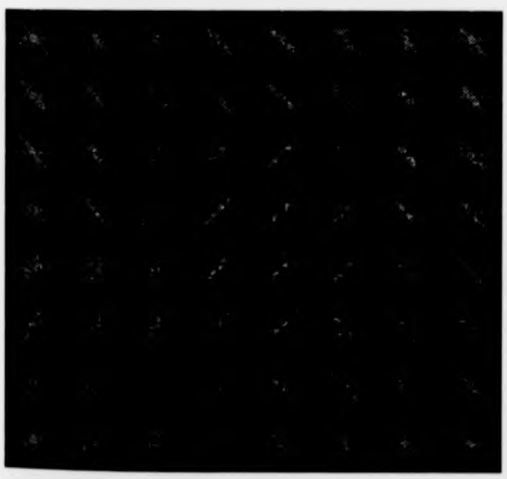
$$y(m,n) = x(m,n) * g(m,n) \tag{3.26}$$

where '$*$' denotes convolution and

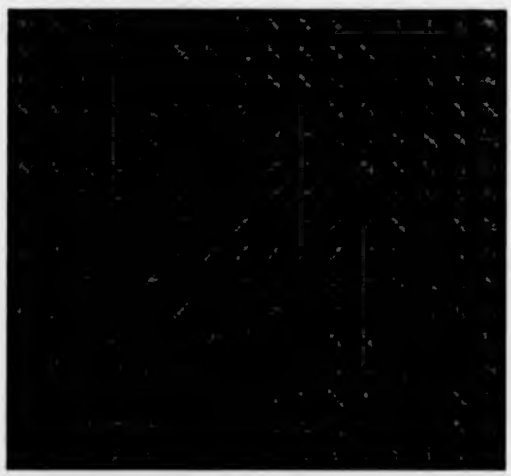
$$g(m,n) = \exp[-[(m\cos\theta - n\sin\theta)^2 + R^2(m\sin\theta + n\cos\theta)^2]] \tag{3.27}$$

(a) Original synthsised texture patches

(b) The DFT of original



(c) The MFT at level 3



(d) The MFT at level 4

Figure 3.13: A synthetic texture field and its MFT's at different resolutions.

By using different values of R, T and θ, this filtering operation captures some of the important features of a texture, such as the granularity and the directionality.

Figure 3.13(a) shows an image containing 3 textures produced by this method and its MFT at 3 levels. As discussed in Chapter 2, the DFT of an image is a global representation which lacks local information about texture features. The DFT of the image (MFT level 0), shown in (b), displays the orientation of the different patches but no boundary information. Figures (c) and (d) show intermediate MFT levels which provide both class and positional information about the textured image at two different resolutions. In particular, there would appear to be sufficient information at level 4 to segment the textures on the basis of magnitude only.

To illustrate the MFT symmetry properties, an FM-Radial test pattern was generated. The FM-Radial test pattern $y(m,n)$ was generated by

$$y(m,n) = f(m,n) + s(m,n) \tag{3.28}$$

where $f(m,n)$ is a radial FM pattern defined as [63]

$$f(m,n) = \begin{cases} 127 & \text{if } r < 56 \\ 127[1 + \text{tr}[\cos^2(\frac{r\pi}{16} - 4\pi)]] & \text{if } 56 \leq r < 64 \\ 127[1 + \text{tr}] & \text{if } 64 \leq r < 224 \\ 127[1 + \text{tr}[\sin^2(\frac{r\pi}{64} - 4\pi)]] & \text{if } 224 \leq r < 256 \end{cases} \tag{3.29}$$

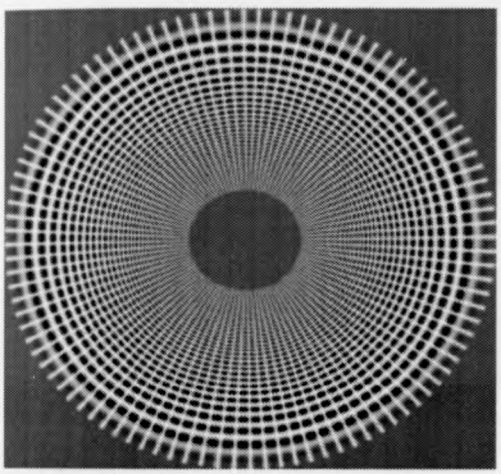
where

$$\text{tr} = \sin(\frac{112\pi}{\log(2)}(2^{(\frac{-r}{56})} - 2^{(\frac{256}{56})})) \tag{3.30}$$

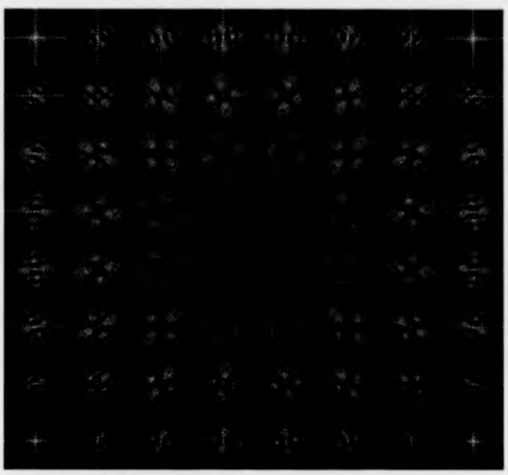
and r is the distance from the centre of the image in pixel distance units and $s(m,n)$ is a sine function defined as

$$s(m,n) = \sin(A \tan^{-1}(\frac{n}{m})) \tag{3.31}$$

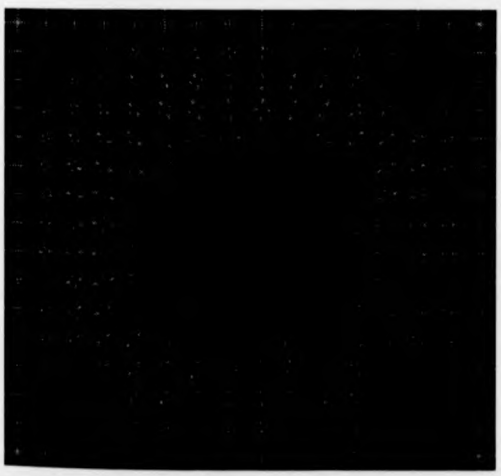
where A is a constant.



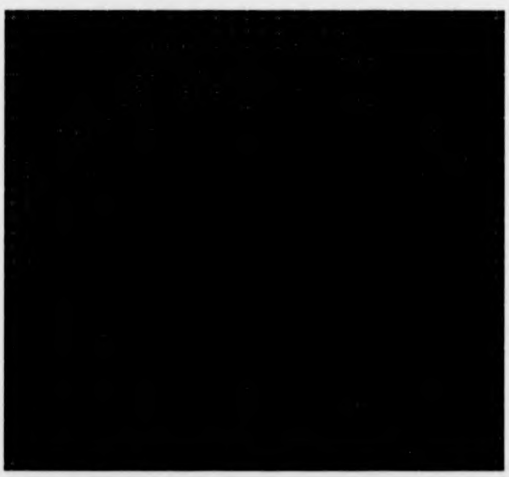
(a) Original FM-Radial test pattern



(b) The MFT at level 3



(c) The MFT at level 4



(d) The MFT at level 5

Figure 3.14: The FM-Radial test pattern and its MFT's at different resolutions.

Figures 3.14(a) shows the synthesised FM-Radial test pattern. Figures (b), (c) and (d) show the change in spatial and spatial frequency resolution between the levels of the transform. The distribution of energy in the MFT local spectra over orientation is perpendicular to that in the spatial image. These figures also show that the scale of the local spectra near the origin of the test pattern increases whilst for those further away from origin the scale decreases. Note also that in this image, the local spectra in each level are approximately related by rotation and scaling.

3.6.2 Natural Textured Images

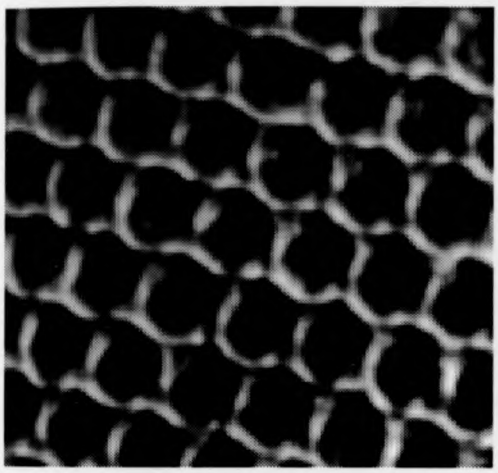
The reptile skin texture taken from [11] demonstrates the capability of the MFT in the description of natural textures, which show significant local structure. Figure 3.15(a) is the reptile skin image of size 128 × 128. Figures (b), (c) and (d) show that the MFT coefficients at appropriate levels reveal both the texture element distribution and its frequency content.

3.7 Conclusion

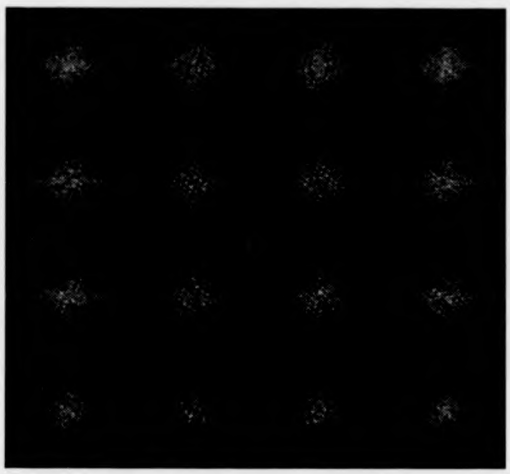
Two different MFT implementations have been presented - the spatial implementation using an index-limited window function and the frequency implementation using a band-limited window function. Both are efficiently implemented using the FFT algorithm. In terms of computational requirements, the two approaches are equivalent. The computational complexity for the complete MFT calculated in terms of complex multiplication O of one dimensional FFT is given as [16]

$$O = \frac{N^2}{2} \left[(\log_2 N)^2 + 3\log_2 N - 2\right] \tag{3.32}$$

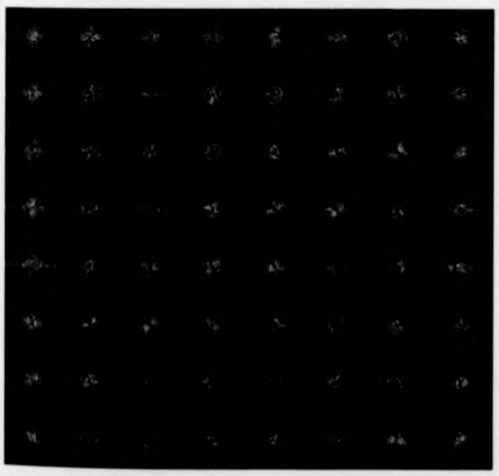
where N is the window size.



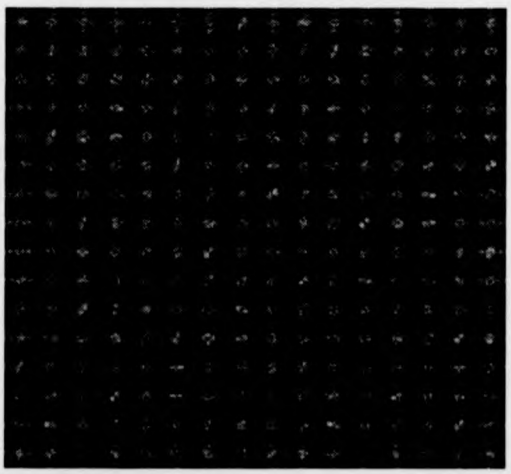
(a) Original Reptile skin image



(b) The MFT at level 2



(c) The MFT at level 3



(d) The MFT at level 4

Figure 3.15: The reptile skin and its MFT's at different resolutions.

Although there is little to choose between the two implementations, it was decided to use the band-limited window and frequency domain implementation in order to minimise interference between frequency domain features, which are essential in texture analysis. The work of Spann and Wilson showed that, even with critical sampling, the boundary resolution obtained using band-limited windows is satisfactory [91].

Chapter 4

Texture analysis and synthesis

4.1 Introduction

It was argued in Chapter 1 that a more general model of texture is needed than either the random placement rule of stochastic models or the regular placement of simple structural models. While the underlying principle of repetition of a prototype is a useful first step, some way must be found of allowing the prototype to vary if textures such as the reptile skin of figure 4.1 are to be modelled. One interesting approach to this is suggested by the 'fractal' methods described by Barnsley [6] and used by Jacquin in image data compression [57]. These are based on affine transformations: patches of an image are related by affine transformations selected from a restricted set. Because they allow for rotation, scaling and shear, 2-D affine transforms can be used to approximate the effects of perspective in computer vision [43]. It therefore seems useful to attempt to model textures in this way: it is an obvious generalisation of periodic models to introduce greater degrees of freedom and hence to extend the structural approach to deal with more realistic textures.

It will be shown in this chapter that the MFT provides an efficient and general way to implement analysis and synthesis based on the affine model, which extends

the earlier work of Volet [97] and of Gårding on the use of texture in 3-D shape estimation [43]. The overall theme is therefore to identify the affine transform between local patches of the image and hence resynthesise the image from a prototype texture element warped using the inverse affine transform. The affine transformations T used in this work are 2-D transformations of the form

$$T\vec{\xi} = A\vec{\xi} + \vec{\gamma} \tag{4.1}$$

where A is an invertible linear transform, $\vec{\gamma}$ is a translation and $\vec{\xi}$ is the spatial co-ordinate.

Consider the case illustrated in figure 4.1, in which 2 regions are selected from the reptile skin image, with one of the local regions a warped version of the other. This warping is modelled by the affine transform. The first problem is then to identify the best fitting affine transform parameters, given the two texture blocks. ie.

$$\begin{aligned} \vec{\xi}_2 &= T_{12}\vec{\xi}_1 \\ &= A_{12}\vec{\xi}_1 + \vec{\gamma}_{12} \end{aligned} \tag{4.2}$$

where $\vec{\xi}_1$ is the co-ordinate in the first block, $\vec{\xi}_2$ is the corresponding co-ordinate in the second, and A_{12} is the linear transform from block 1 to block 2, $\vec{\gamma}_{12}$ is the relative spatial translation between them.

4.2 Identification of Affine Transform

This technique is based on the local Fourier spectra of the prototype and texture element of interest, calculated using the MFT. Recall from Chapter 2 that a given level l of the MFT may be described as a set of estimated local spectra and is defined

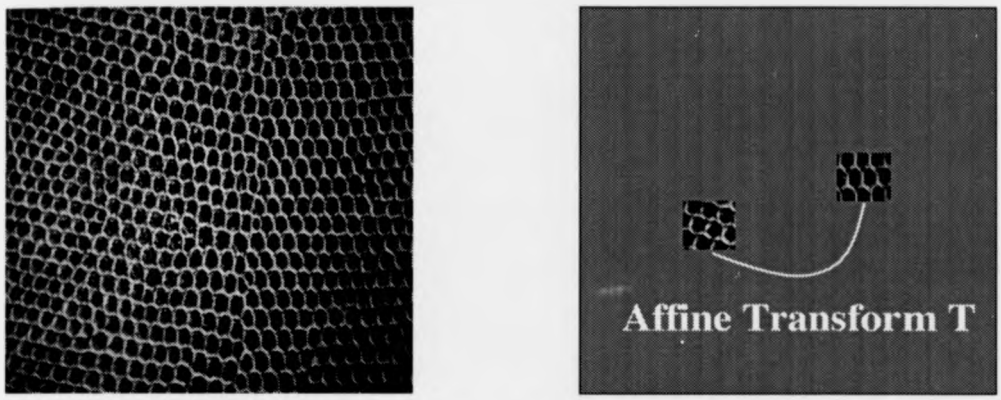


Figure 4.1: Texture warping is modelled via affine transform.

as [16] [30]

$$\hat{x}(\vec{\xi}_i(l), \vec{\omega}_j(l), l) = \sum_k w_l(\vec{\xi}_i(l) - \vec{\xi}_k) x(\vec{\xi}_k) \exp[-j\vec{\omega}_j(l).\vec{\xi}_k] \tag{4.3}$$

for notational simplicity, this is represented by $\hat{x}(\vec{\omega}_j)$.

An MFT block at a given level l representing a local Fourier spectrum is first selected as the prototype at that scale. The affine transform of the prototype to each of the remaining MFT blocks is identified. On the basis of the identified transform, the original texture image can be resynthesised.

The implementation consists of three main components:

1. Determination of the linear co-ordinate transformation A_{12} which gives the best fit between the magnitude spectra.

2. Linear transformation of the prototype spectrum to give an approximation to that of the texture element.

3. Correlation of the texture element and transformed prototype by multiplication of the respective spectra and inverse Fourier transformation, giving a 'goodness

of fit' measure and an estimate of the spatial translation using the peak of the correlation.

Figure 4.2 illustrates the main components of the algorithm. For two given local spectrum estimates $\hat{x}^{(1)}(\vec{\omega}_j)$ and $\hat{x}^{(2)}(\vec{\omega}_j)$, corresponding to the two local regions of the input image $x^{(1)}(\vec{\xi}_i)$ and $x^{(2)}(\vec{\xi}_i)$, the centroid vector pair is used to represent the spectrum; from the centroid vector pair, a linear transform A is identified. Based on the linear transform A, the prototype spectrum $\hat{x}^{(2)}(\vec{\omega}_j)$ is transformed to $\hat{x}^{(2)'}(\vec{\omega}_j)$. At this stage, the magnitude spectrum $\hat{x}^{(2)'}(\vec{\omega}_j)$ is an approximation of the test spectrum $\hat{x}^{(1)}(\vec{\omega}_j)$ except for the translation, which appears in the Fourier transform phase. The inner product and inverse DFT are then applied, effectively performing a spatial correlation.

The basis of the approach is therefore to split the identification of the affine transformation into two parts: determination of its linear component by mapping the shift invariant magnitude spectra; and subsequent determination of the translation component using correlation. This has the advantage of being both computationally efficient and automatically providing a certainty measure of the match between the two, thus yielding a simple means of assessing the suitability of the model. Although it might be possible in principle to identify all 6 parameters of the affine transform by a search, the computation required would render such a method infeasible.

4.2.1 Representative Centroid Vectors

A pair of representative centroid vectors is extracted from a given local magnitude spectrum as the representation of the spectrum. The estimate of the representative centroid vectors corresponding to the linear component of the affine transform is

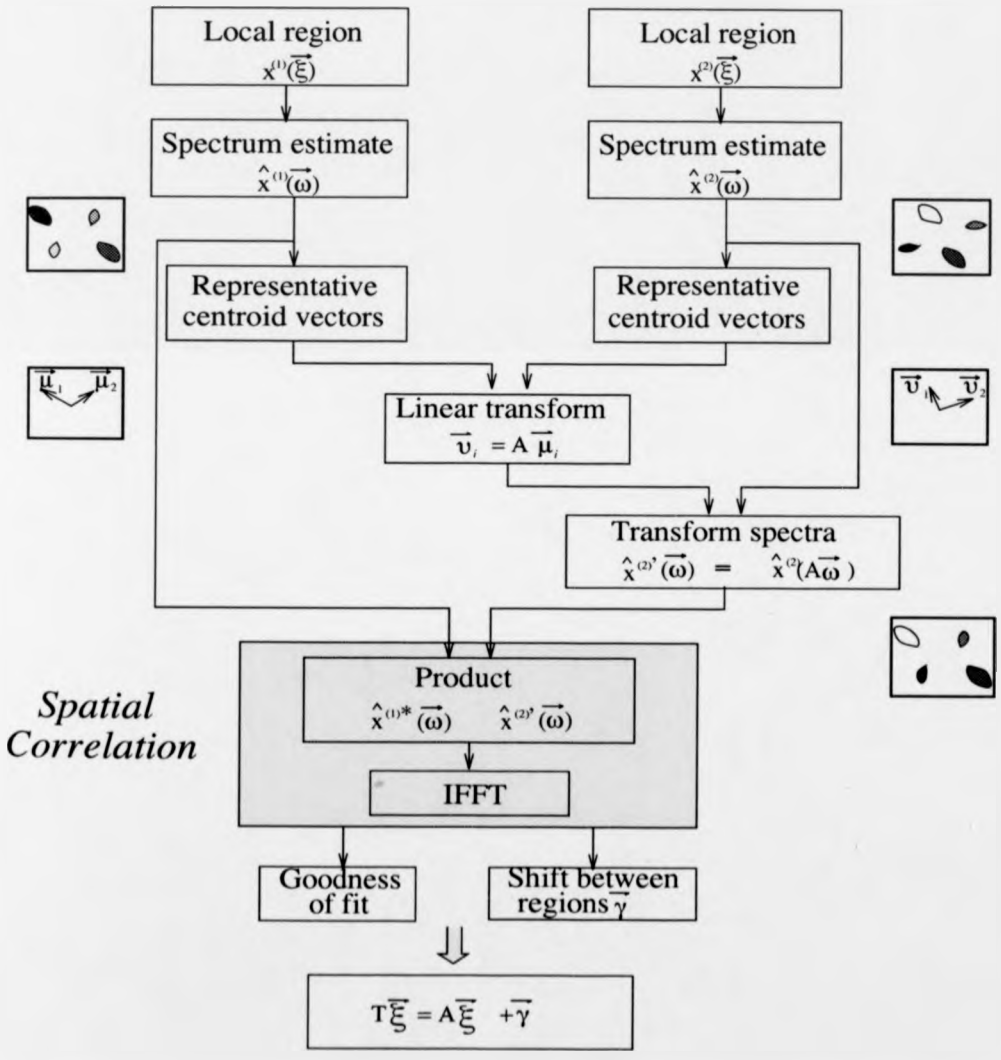


Figure 4.2: Outline of algorithm.

computed on the basis of the angular frequency spectral energy distribution. Taking advantage of the Hermitian symmetry of the Fourier transform [75], the computational requirement to estimate the centroid vectors in the frequency domain is only half the number of points in the MFT block, $\hat{x}(\vec{\omega}_j)$.

Given a discrete Fourier spectrum $\hat{x}(\vec{\omega}_j)$ corresponding to a prototype or texture element at a given scale, the procedure is to find the angles θ_1 and θ_2 such that the sum of the variances

$$\sum\nolimits^2(\theta_1,\theta_2) = \sigma_1^2(\theta_1,\theta_2) + \sigma_2^2(\theta_1,\theta_2) \tag{4.4}$$

is a minimum, where

$$\sigma_i^2(\theta_1,\theta_2) = \frac{1}{M_i(\theta_1,\theta_2)} \sum_{\vec{\omega}_j \in \Lambda_i(\theta_1,\theta_2)} |\hat{x}(\vec{\omega}_j)| \;\; \| \vec{\omega}_j - \vec{\mu}_i(\theta_1,\theta_2) \|^2 \qquad i = 1,2 \tag{4.5}$$

$$M_i(\theta_1,\theta_2) = \sum_{\vec{\omega}_j \in \Lambda_i(\theta_1,\theta_2)} |\hat{x}(\vec{\omega}_j)| \tag{4.6}$$

$\vec{\mu}_i(\theta_1,\theta_2)$ is the centroid

$$\vec{\mu}_i(\theta_1,\theta_2) = \frac{1}{M_i(\theta_1,\theta_2)} \sum_{\vec{\omega}_j \in \Lambda_i(\theta_1,\theta_2)} |\hat{x}(\vec{\omega}_j)| \, \vec{\omega}_j \tag{4.7}$$

and $\Lambda_1(\theta_1,\theta_2)$ and $\Lambda_2(\theta_1,\theta_2)$ are the sets of coordinates in each segment of the half-plane at an angle θ_1 and divided at an angle θ_2, ie

$$\Lambda_1(\theta_1,\theta_2) = \Lambda(\theta_1) - \Lambda(\theta_1+\theta_2) \cap \Lambda(\theta_1) \tag{4.8}$$

$$\Lambda_2(\theta_1,\theta_2) = \Lambda(\theta_1) \cap \Lambda(\theta_1+\theta_2) \tag{4.9}$$

where $\Lambda(\theta)$ denotes the set of half-plane coordinates at an angle θ, i.e.

$$\omega_{j2}\cos\theta - \omega_{j1}\sin\theta \;\geq\; 0 \qquad \forall \;\; \vec{\omega}_j \;\in\; \Lambda(\theta). \tag{4.10}$$

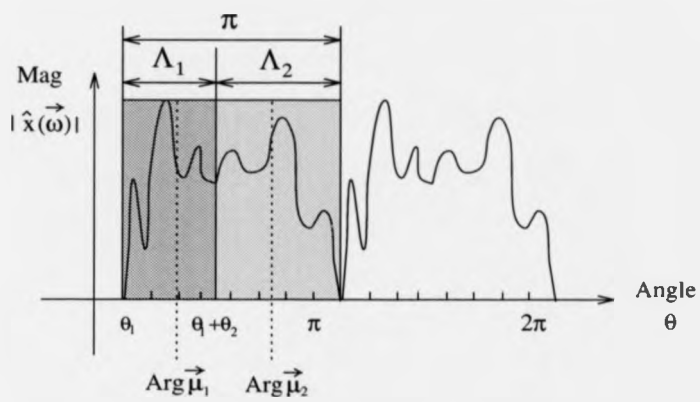


Figure 4.3: Representative centroid vectors; Λ_1 is coordinates between $(\theta_1, \theta_1 + \theta_2)$, Λ_2 is coordinates between $(\theta_1 + \theta_2, \theta_1 + \pi)$, $\vec{\mu_1}$ is centroid in Λ_1 and $\vec{\mu_2}$ is centroid in Λ_2.

In figure 4.3, for a given local spectrum, the magnitude is plotted versus the angle θ. Due to the symmetry of the spectrum, the estimation is performed in the half plane. θ_1 is used to split the local spectrum into two halfplanes and θ_2 further splits one of the halfplanes into two segments (Λ_1, Λ_2). In each segment, the centroid and variance and sum of the two variances is computed. Each (θ_1, θ_2) corresponds to one pair of centroid vectors and variance sum $\sum^2(\theta_1, \theta_2)$. The angles (θ_1, θ_2) which minimise the sum of variances are chosen and the corresponding centroids are chosen as the representative vectors.

In general, the distribution of the local spectrum corresponding to a given texture has various patterns and contains different numbers of energy peaks. No matter how many there are, this method of identifying the centroid vectors can cope well, as the examples shown in figure 4.4 illustrate. In case 1 there are 2 clusters and the

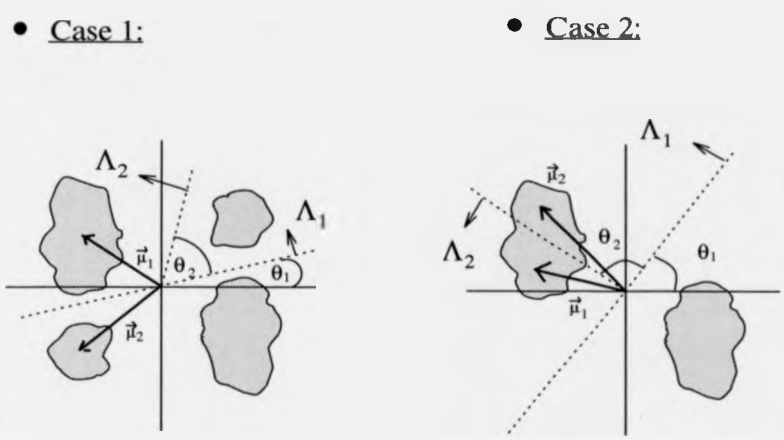


Figure 4.4: Modelling general cases with different numbers of clusters.

estimate of the centroid vector associated with minimum variance sum can efficiently represent the local spectrum. For case 2 there exists only 1 cluster, but the extracted representative centroid vectors are still capable of describing the local spectrum. For the case of more than 2 clusters, this model can still be applied. The reason for finding 2 vectors is that 4 is the minimum number of parameters needed to derive the linear coordinate transform. Moreover, the use of centroids based on partitions along lines through the origin makes the choice robust to the transformations which are being estimated : the linear transform A transforms any line through the origin into another such line.

In order to perform the calculations, each distinct combination of angles $\theta_1 < \theta_2 \leq \pi$, where $\theta_i = \pi k/N$, is tested, as shown in figure 4.5. This shows the variance sum as a function of θ_1 and θ_2 on a 3-D surface plot, for one block selected from the reptile skin. In this example, the minimum variance sum occurred at $(\theta_1, \theta_2) = (2.67, 1.178)$ in radian unit.

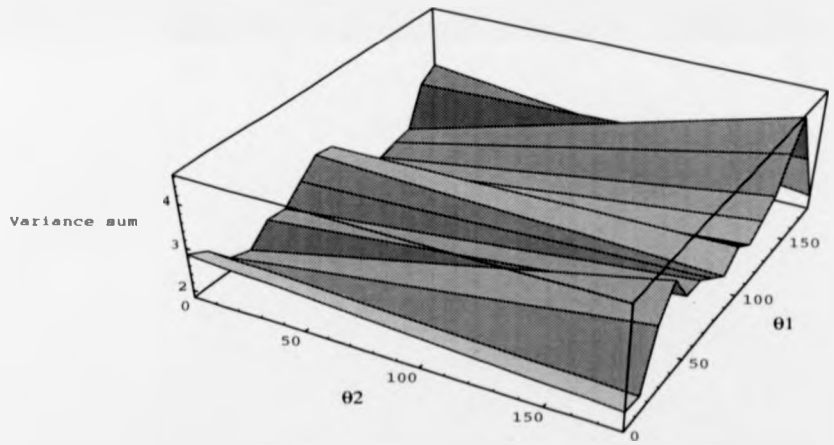


Figure 4.5: Variance sum as a function of θ_1 and θ_2 in degrees.

4.2.2 Estimation of Linear Transform

The algorithm proceeds by obtaining the transform matrix relating the two pairs of centroid vectors. The symmetry of the spectra means that there are eight possible linear transformations that need to be tested. This in turn leads to the 2×2 matrices $A_1 - A_8$, as shown in figure 4.6. The matrix A is defined as in equation (4.11) such that $\vec{\mu}_i^{(1)} = A\vec{\mu}_i^{(2)}$, $i = 1, 2$.

$$\begin{bmatrix} \vec{\mu}_1^{(1)} \\ \vec{\mu}_2^{(1)} \end{bmatrix} = A \begin{bmatrix} \vec{\mu}_1^{(2)} \\ \vec{\mu}_2^{(2)} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} \vec{\mu}_1^{(2)} \\ \vec{\mu}_2^{(2)} \end{bmatrix} \tag{4.11}$$

A_1 is formed by calculating from centroid pair $(\vec{\mu}_1^{(2)}, \vec{\mu}_2^{(2)})$ to $(\vec{\mu}_1^{(1)}, \vec{\mu}_2^{(1)})$. A_2 is obtained from A_1 by swapping centroid pair from $(\vec{\mu}_1^{(2)}, \vec{\mu}_2^{(2)})$ to $(\vec{\mu}_2^{(2)}, \vec{\mu}_1^{(2)})$. A_3 is formed by replacing $\vec{\mu}_2^{(2)}$ by $-\vec{\mu}_2^{(2)}$. In A_4, the centroid pair is swapped from that of A_3 in the same way as the A_2 vectors are swapped from A_1. A_j, $j = 5, ..., 8$ are obtained from A_k, $k = 1, ..., 4$ respectively by negating $(\vec{\mu}_1^{(2)}, \vec{\mu}_2^{(2)})$.

To determine which of these gives the best fit, a 'synthesis and correlation' method is adopted: for each case, the prototype is transformed and correlated with the original test texture element to find the best match. Thus, for each of the matrices $A_1 - A_8$, the prototype spectrum $\hat{x}^{(2)}(\vec{\omega}_j)$ is transformed according to

$$\hat{x}_i^{(2)'}(\vec{\omega}_j) = \hat{x}^{(2)}(A_i\vec{\omega}_j) \qquad 1 \leq i \leq 8 \tag{4.12}$$

where a given coefficient $\hat{x}_i^{(2)'}(\vec{\omega}_j)$ is calculated from the discrete coefficients $\hat{x}^{(2)}(\vec{\omega}_j)$ using bilinear interpolation from its four nearest neighbours shown in figure 4.7. Let (u, v) be the real number coordinate whose four nearest integer coordinate points are (ω_u, ω_v), $(\omega_u + 1, \omega_v)$, $(\omega_u, \omega_v + 1)$ and $(\omega_u + 1, \omega_v + 1)$ shown in figure 4.8, then the

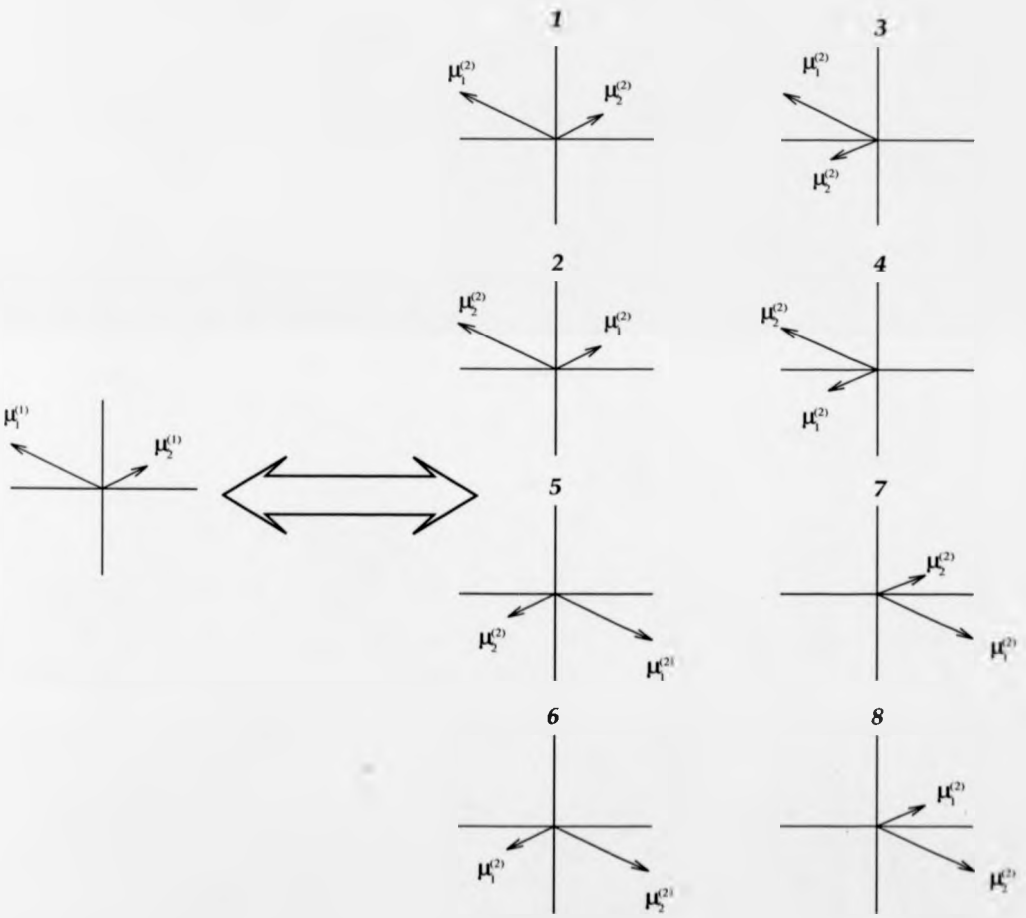


Figure 4.6: Eight possible transform matrices.

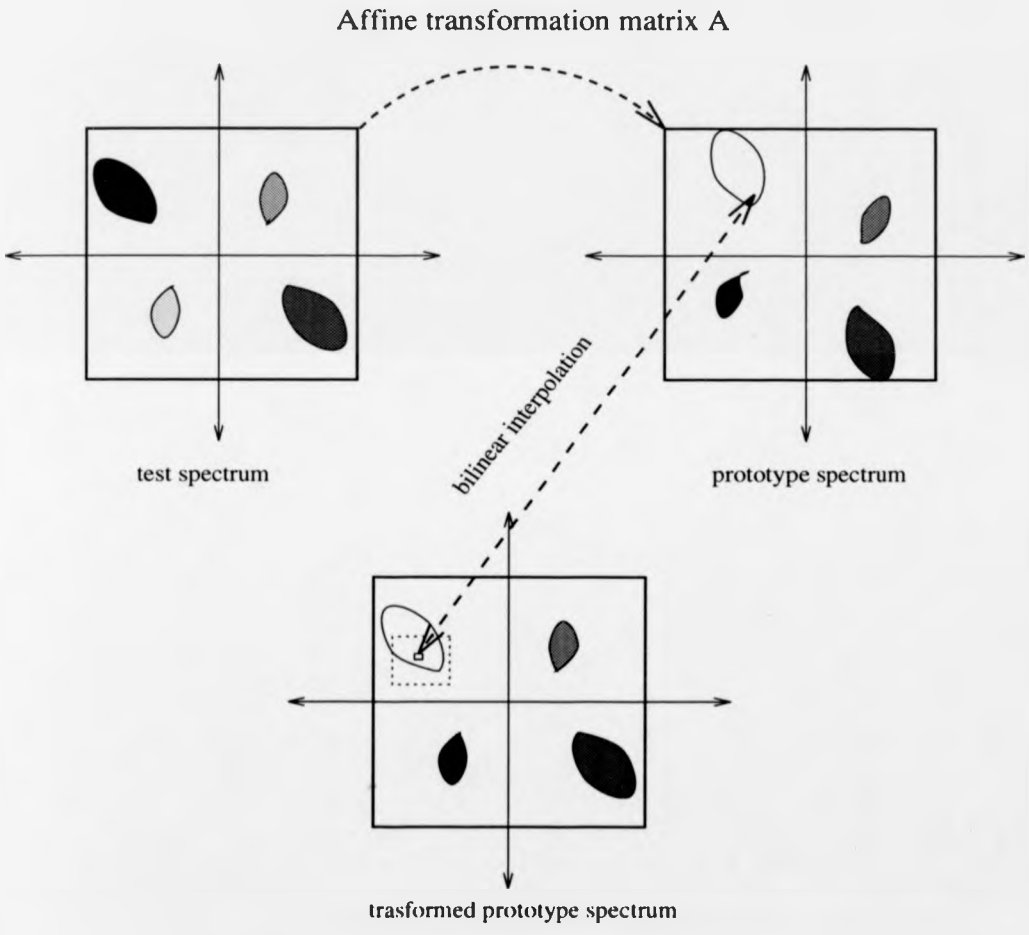


Figure 4.7: Relationship between prototype, test spectrum and transformed prototype (Note: transformed magnitude spectrum is similar to test's except phase).

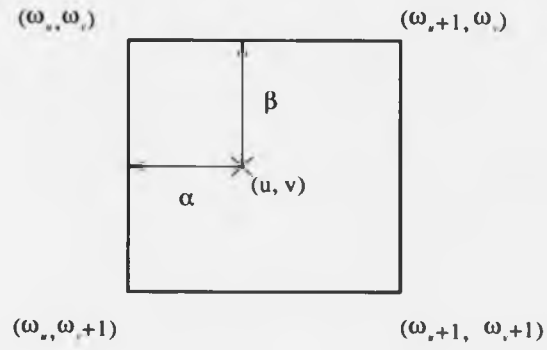


Figure 4.8: Bilinear interpolation.

bilinear interpolation estimate $\hat{x}^{(2)}(u,v)$ is given as :

$$\begin{aligned}\hat{x}^{(2)}(u,v) = & \\ & (1-\alpha)(1-\beta)\hat{x}^{(2)}(\omega_u,\omega_v) \\ & +\alpha(1-\beta)\hat{x}^{(2)}(\omega_u,\omega_v+1) \\ & +(1-\alpha)\beta\hat{x}^{(2)}(\omega_u+1,\omega_v) \\ & +\alpha\beta\hat{x}^{(2)}(\omega_u+1,\omega_v+1) \end{aligned} \tag{4.13}$$

where

$$\begin{aligned}\alpha &= u-\omega_u, \\ \beta &= v-\omega_v \end{aligned} \tag{4.14}$$

It was shown by Calway [16] that this interpolation is adequate for estimating MFT coefficients at arbitrary co-ordinates $\vec{\omega}$, given the sampling scheme used in this

work.

As described in figure 4.7, the transformed prototype spectrum is the linear coordinate transformed approximation of the prototype; in the illustrated case, $\hat{x}_i^{(2)'}(\vec{\omega}_j)$ is the rotated version of $\hat{x}^{(2)}(\vec{\omega}_j)$.

4.2.3 Correlation and Best Fit

The goodness of fit between the synthesised texture element and the actual texture element is measured by calculating the spatial correlation function. By applying the corresponding inner product between spectra directly in the frequency domain, the correlation between two spatial regions can be computed efficiently. Hence, the transformed prototypes are correlated with the test texture element using normalised correlation, which is performed directly in the frequency domain by applying the energy theorem [75]

$$\rho_i(\vec{\xi}_j) = \frac{F^{-1}\{\hat{x}_i^{(2)'}(\vec{\xi}_j, \vec{\omega}_k)\hat{x}^{(1)*}(\vec{\xi}_j, \vec{\omega}_k)\}}{\sum_k |\hat{x}_i^{(2)'}(\vec{\xi}_j, \vec{\omega}_k)|\,|\hat{x}^{(1)}(\vec{\xi}_j, \vec{\omega}_k)|} \tag{4.15}$$

where * is the complex conjugate, F^{-1} denotes the inverse discrete Fourier transform (DFT), and $\vec{\xi}_j$ is the spatial coordinate vector. The largest correlation peak then determines the selected transformation, i.e. the best match is achieved by transforming the prototype according to the matrix A_k such that

$$\epsilon = \rho_k(\vec{\xi}_l) = \max_j \rho_k(\vec{\xi}_j) = \max_i (\max_j \rho_i(\vec{\xi}_j)) \tag{4.16}$$

where ϵ is the 'goodness of fit' measure. As an illustration, figure 4.9 shows the search for the peak value in the correlation fields determining the best fit, where $\vec{\xi}_l$ gives the phase shift for reconstruction (cf 4.3).

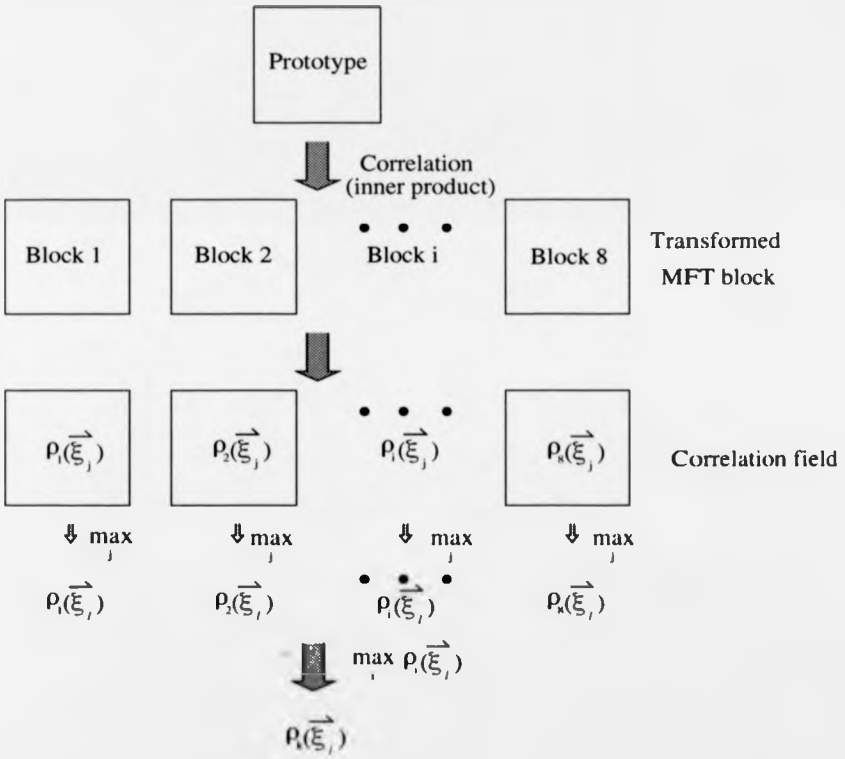


Figure 4.9: Correlation between prototype and the 8 transformed MFT blocks gives the best fit.

4.3 Image Synthesis and Reconstruction

It was mentioned in the previous section that affine transform is a means of description of the textural structure variation. Hence, following the identification of the best fit affine transform at a given level, the synthesis of all other blocks from the prototype is integrated to give a synthesised image.

After performing normalised correlation in the multiresolution transform, the image synthesis is accomplished by the following stages:

1. *Phase shift*: the phase shift information between the test spectrum and the best transformed spectrum is provided by the correlation calculation. This phase shift corresponds to the spatial displacement between the two signals. To correct the position of the transformed spectra, the phase term is included prior to the inverse DFT.

$$\hat{x}_k^{(2)}(\vec{\omega}_l) = \hat{x}_k^{(2)'}(\vec{\omega}_j)e^{-j\vec{\omega}_j.\vec{\xi}_l} \tag{4.17}$$

   where $j^2 = -1$ and $\vec{\xi}_l$ is defined in (4.16).

2. *Synthesis of the texture element*: after phase correction (4.17), the synthesised texture element is given by

$$\hat{x}_k^{(1)}(\vec{\xi}_j) = F^{-1}\{\hat{x}_k^{(2)'}(\vec{\omega}_l)\} \tag{4.18}$$

3. *Energy normalisation*: energy normalisation is used to keep the energy of the synthesised block the same as that of the target block, ie.

$$\tilde{x}_k^{(1)}(\vec{\xi}_j) = \alpha\hat{x}_k^{(1)}(\vec{\xi}_j) \tag{4.19}$$

where

$$\alpha^2 = \frac{\sum_j |\hat{x}^{(1)}(\vec{\omega}_j)|^2}{\sum_j |\hat{x}_k^{(2)}(\vec{\omega}_j)|^2} \tag{4.20}$$

As a result of the energy normalisation, the synthesis has the same intensity variation as that of the original image.

4. *Cosine windowing and integrating*: the output of the inverse Fourier transform is twice the size of the nominal spatial block because the MFT is oversampled [16]. It is therefore possible to invert the transform by either taking the middle part of the block, which will introduce blocking artifacts, or performing windowing with, e.g. a raised cosine function as shown in (4.21), over the inverse Fourier transform output followed by adding them up to give a synthesised image without blocking artifacts. Considering the one dimensional case in figure 4.10, the reconstructed signal from 50% overlapping windows can be achieved by applying the cosine squared window on the interval of interest. In this case, for two adjacent windows considering the same coordinate $\vec{\xi}_j$, one particular spatial value at coordinate $\vec{\xi}_j$ within that interval is effectively reconstructed from

$$\begin{aligned}
\hat{x}(\vec{\xi}_j) &= \hat{x}_k^{(1)}(\vec{\xi}_j)(cos^2(\pi(\vec{\xi}_j + nN + N/2)/N) + cos^2(\pi(\vec{\xi}_j + (n+1)N)/N)) \\
&= \hat{x}_k^{(1)}(\vec{\xi}_j)(cos^2(\pi\vec{\xi}_j/N + \pi/2) + cos^2(\pi\vec{\xi}_j/N)) \\
&= \hat{x}_k^{(1)}(\vec{\xi}_j)(sin^2(\pi\vec{\xi}_j/N) + cos^2(\pi\vec{\xi}_j/N)) \\
&= \hat{x}_k^{(1)}(\vec{\xi}_j)
\end{aligned} \tag{4.21}$$

where $\hat{x}_k^{(1)}(\vec{\xi}_j)$ is defined as in equation (4.17).

5. *Lowpass superposition*: because the energy of the lowpass portion of the MFT coefficient dominates the spectrum, only the high pass spectrum is used in the

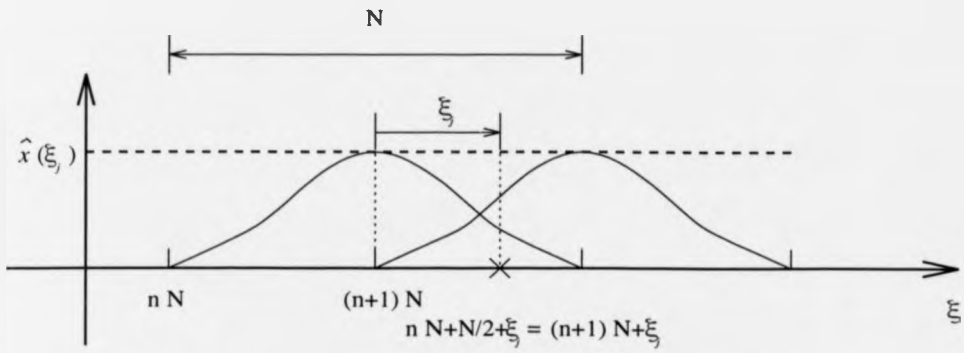


Figure 4.10: Reconstruction from the cosine square function.

texture analysis and synthesis. Prior to the identification of the affine transforms, the image is high pass filtered by removing the lowpass component. At the end of the synthesis process, the low pass portion is reconstructed by padding a number of zeros up to the same size of the lowpass image followed by the inverse DFT to obtain a smoothed lowpass image. The reconstructed low pass image is then combined with the synthesis result from the high pass spectrum to obtain the final result. Figure 4.11 shows this process. The reconstructed lowpass image of the reptile skin at the MFT level 5 is shown in figure 4.12.

4.4 Synthesis Fidelity Criterion

Because information of interest in the original texture may be lost during both the transform and interpolation procedure, the *mean-square signal-to-noise ratio* of the synthesised result, denoted by SNR_{ms}, is used as the basis for synthesis fidelity. For the input texture image $x(\vec{\xi}_j)$ of size $N \times N$ pixels and the synthesised result $\hat{x}(\vec{\xi}_j)$,

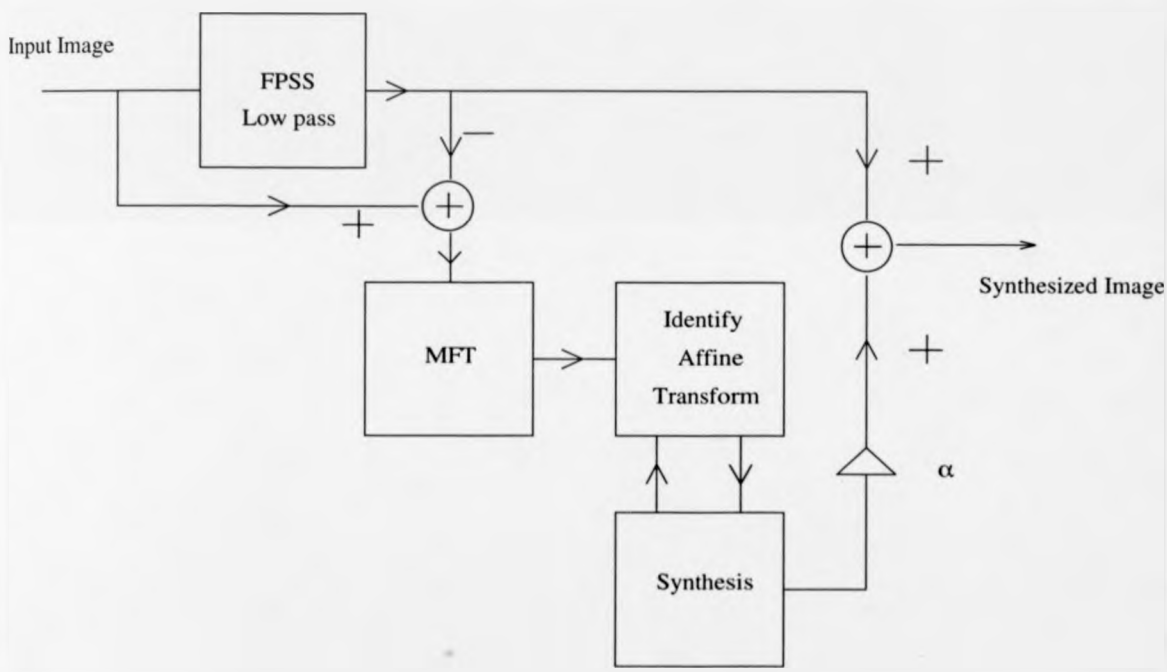


Figure 4.11: Low pass portion removed in the beginning and combined with synthesis result in the end, where α is the energy factor.

Figure 4.12: Lowpass image of the reptile skin reconstructed at MFT level 4.

the SNR_{ms} is given as

$$SNR_{ms} = 10\log_{10}\left(\frac{1}{\epsilon(\vec{\xi}_j)}\sum_{\vec{\xi}_j}(x(\vec{\xi}_j)-\mu)^2\right) \tag{4.22}$$

where

$$\epsilon(\vec{\xi}_j) = \sum_{j}(x(\vec{\xi}_j)-\hat{x}(\vec{\xi}_j))^2 \tag{4.23}$$

and μ is the mean luminance

$$\mu = \frac{1}{N^2}\sum_{j} x(\vec{\xi}_j). \tag{4.24}$$

Prior to calculation of the SNR_{ms}, the variation of the magnitude in the synthesised image is often different from that in the original image because of the changes of the first and second order statistics. Therefore, the comparison should be made on the basis that the two images have the same mean value and variance. The synthesised image is scaled to achieve this.

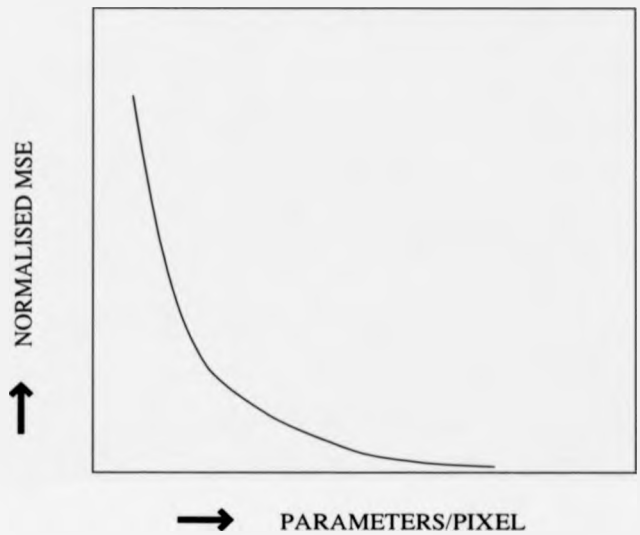


Figure 4.13: Relationship between the quality of synthesised image and the number of parameters used per pixel for synthesis.

4.5 Choice of Synthesis Scale

The choice of synthesis scale implies a trade-off between the accuracy of synthesis results and number of parameters per pixel being used for synthesis. This is a form of rate-distortion problem [88]. The quality of an image synthesised and the number of parameters used per pixel for synthesis depend on the following parameters: the better spatial resolution of the MFT coefficients, the better the quality of the synthesised image, but higher the information rate. Figure 4.13 shows the nature of this relationship. The accuracy of the synthesis is directly related to the number of parameters to transform one MFT block. Each affine transformation requires 6 parameters: 4 for linear transform components, 2 for the translation. In addition, the subsampled lowpass image requires 4 samples per MFT block and 1 parameter

for energy normalisation per MFT block is required for synthesis. This gives a total of 11 parameters per MFT block. For an image size of $N \times N$ pixels and MFT block size of $M_l \times M_l$ at a given level l, the total number of parameters in the synthesis is $11 \times \frac{(2N)^2}{M_l^2}$, where $2N$ is the number of samples along each axis obtained from the oversampling scheme. Moreover, the number of samples per pixel in the prototype spectrum is $\frac{M_l^2}{N^2}$. Therefore, the number of parameters per pixel to synthesise an image is

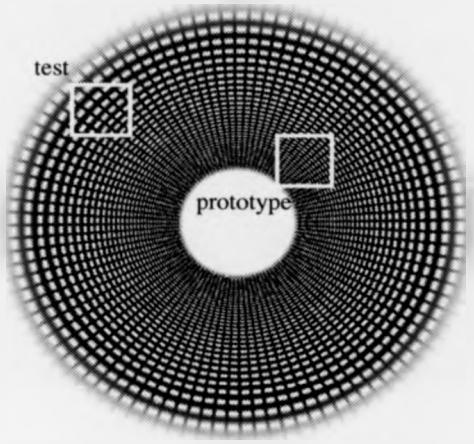
$$\frac{\text{parameters}}{\text{pixel}} = 11 \times \frac{(2N)^2}{M_l^2 \times N^2} + \frac{M_l^2}{N^2} = \frac{44}{M_l^2} + \frac{M_l^2}{N^2} \tag{4.25}$$

If M_4 is twice M_5, the total number of parameters in synthesis at level 5 is four times that at level 4. Hence, if the accuracy of synthesis result at level 4 is within some range, then the compactness of description at level 5 is worse, although the accuracy of the synthesis result at level 5 is better than that at level 4.

The choice of scale is therefore a trade-off between the quality of the synthesised image and the number of parameters used per pixel, which is based on the 'compactness of description' ratio (4.25). The greater is the regularity of the texture, the higher will be the synthesis SNR at a given scale.

4.6 Experimental Results

Various experiments have been performed to illustrate the different stages within the algorithm. To test the algorithm, experiments were first performed on the FM-Radial pattern shown in figure 4.14. This FM-Radial pattern was generated to combine both *radial* and *angular* functions. It shows the characteristics of scaling, rotation and translation which the local affine transform can model effectively.



FM-Radial Test Pattern

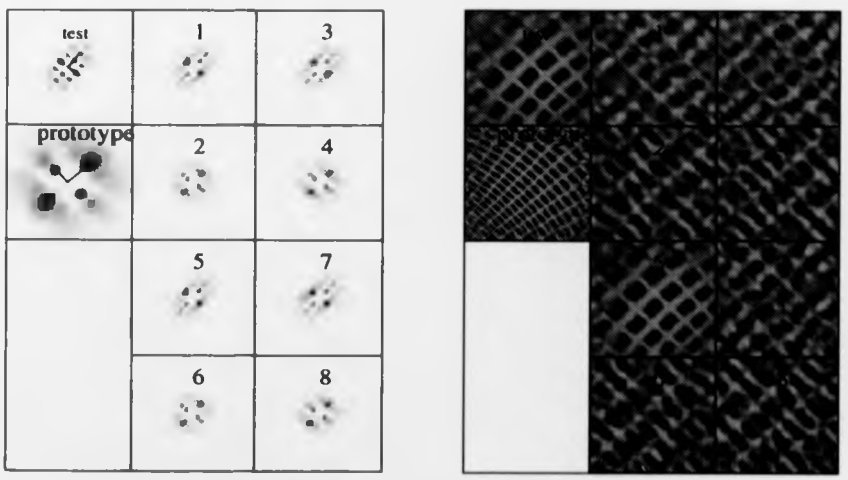


Figure 4.14: FM-Radial Test Pattern and vector pair and the 8 possible spectra from the test block. The resynthesised prototype for each case is shown at the bottom right. Transform number 5 provides the best fit.

4.6.1 Representative Centroid Estimation

A test of the estimate of the representative centroid vectors was performed on the FM-Radial pattern. At the top of figure 4.14, two local regions of the FM-Radial test pattern are shown, while at the bottom of figure 4.14, the representative centroid vectors are extracted and the best affine transform is identified from among the 8 possibilities. Note that the squares indicated do not imply the use of rectangular windowing functions - they indicate the size of the central part of the FPSS spatial windows used in the MFT. According to the correlation between the test spectra and each of the 8 transformed blocks, number 5 block has the highest normalised correlation coefficient (0.965). Therefore, the relationship in both scaling and rotation between the selected prototype and test magnitude spectra can best be described by means of the linear transform matrix 5. When this is combined with the phase shift from the correlation, the desired affine transform is found.

4.6.2 Texture Image Synthesis

To test the performance of the matching algorithm, resynthesis experiments based on both this test pattern and natural images were performed, in which a given texture was generated from an arbitrarily chosen prototype block at various scales. Using the appropriate local spectra, each texture block was synthesised from the prototype block using the procedure detailed above and the blocks combined to give the final synthesis.

As an illustration, figure 4.15 shows for a suitable prototype the affine transforms T1, T2 and T3 which are identified according to the algorithm described above. Texture synthesis is accomplished from the prototype by integrating the individual

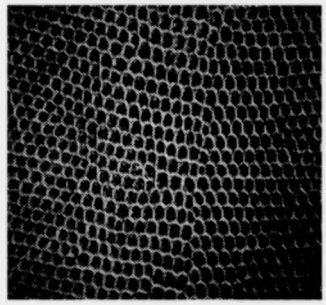
synthesised texture elements based on the identified affine transforms and using the 'overlap-add' method of synthesis described in section 4.3.

Figures 4.16(a) and (b) depict the original FM-Radial test pattern and the synthesis at MFT level 5. Figure 4.16(b) shows that the regularity of the texture and the rotation and scaling of texture elements generated from the radial and angular function are incorporated quite well. Note that in figure 4.16(b) the blur is caused by the bilinear interpolation.

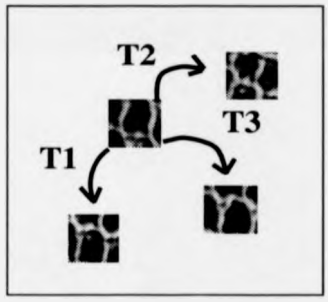
The test images also include three different types of natural texture: reptile skin, burlap and grass from Brodatz's album [11]. Results of experiments for these test images at different scales are shown in figures 4.17,4.18, and 4.19 respectively.

Figures 4.17(a) (b) (c) and (d) depict a reptile skin image size of 512×512 and its synthesis at three different levels. For all of these synthesised results, the regularity and the warping of the texture element in the reptile skin image are resynthesised satisfactorily in comparison with the original image. In figure 4.17(d), there are some artifacts at level 4 due to the scale of the prototype block being significantly larger than the reptile skin texture element. Alignment of the structure between the synthesised texture blocks is missed. Consequently, the variation of the synthesis texture at level 4 is of too low a bandwidth. On the other hand, the result at level 5 in figure 4.17(c) is satisfactory in terms of the SNR listed in Table 4.1 since the scale of the prototype block suits that of the reptile skin texture element. Figure 4.17(b) shows the result at the level 6, which has a good result as well, but, its performance is not good in terms of the compactness of description.

Figure 4.18 depicts a burlap image size of 256×256 and its synthesis at three different scales. Since the size of the texture element of this test image is smaller than



original

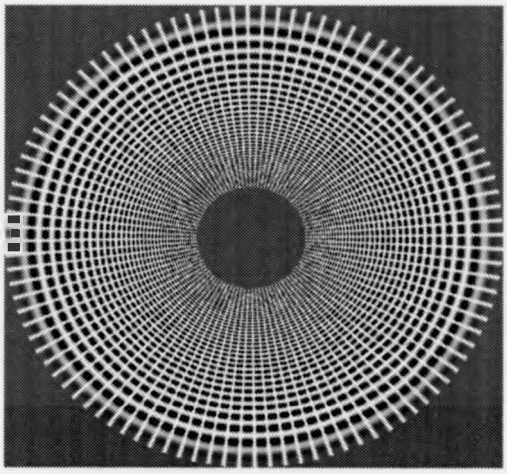


primitive generation from prototype

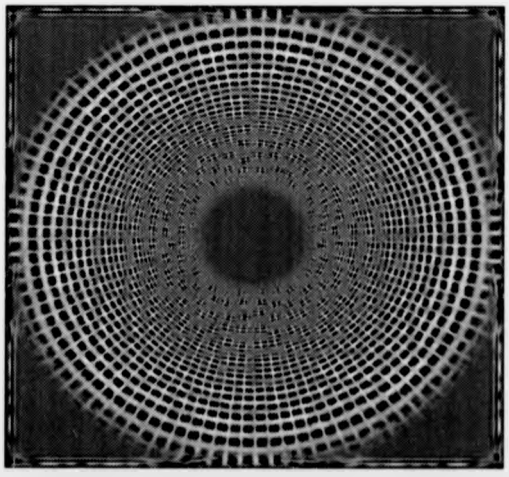
synthesis level 6

synthesis level 5

Figure 4.15: Natural textured image synthesised from a prototype.



(a) FM-Radial test pattern



(b) Synthesis of FM-Radial test pattern at level 5

Figure 4.16: FM-Radial test pattern and synthesis.

MFT Scale Index	Parameters Per pixel	SNR(dB)		
		Reptile	Burlap	Grass
7	0.68774	8.64229	4.04274	4.06653
6	0.17285	7.11266	3.72840	2.07278
5	0.04687	4.76636	1.79312	-0.42035
4	0.02636	2.98247	-0.21225	-1.64412

Table 4.1: SNR of original to synthesised texture for various scales.

that of the reptile skin, the prototype block must be smaller to obtain satisfactory SNR. The synthesis results demonstrate that whatever the size of the texture element, this technique still can perform satisfactorily, provided an appropriate scale is selected. In particular, the dislocation in the middle of burlap image is resynthesised. The substantial effect of the prototype scale is again shown in figure 4.18(d), where the regularity has been increased so that the dislocation becomes less pronounced.

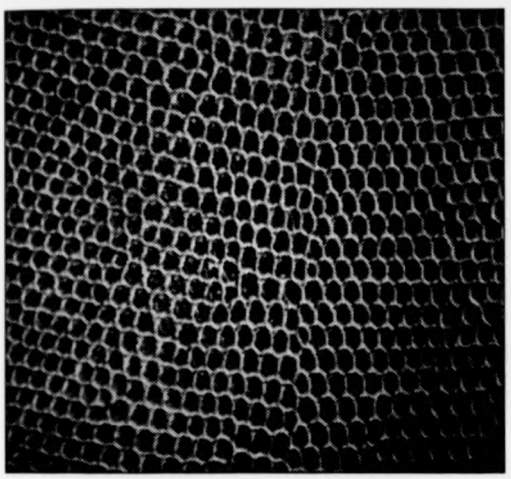
Figure 4.19 shows a grass image of 512 × 512 pixels and its synthesis at three different levels. The synthesised grass image is not as good as that of reptile and burlap in terms of the signal to noise ratio. This is because the randomness of the grass texture significantly affects the accuracy of the synthesis. Nevertheless, the result is reasonable, as shown in figure 4.19(b), which is synthesised at MFT level 7.

Table 4.1 lists the number of parameters used per pixel at each scale and the corresponding SNR of the synthesised texture as in (4.22). Note that the use of smooth, overlapping windows has prevented blocking artifacts. Moreover, the scale at which a given SNR is achieved can be seen from Table 4.1 to depend on the regularity of the texture, as might be expected. In figure 4.20, the normalised mean-square error between an original image and its synthesis is plotted as a function of the number of parameters per pixel used to synthesise it. For an acceptable normalised

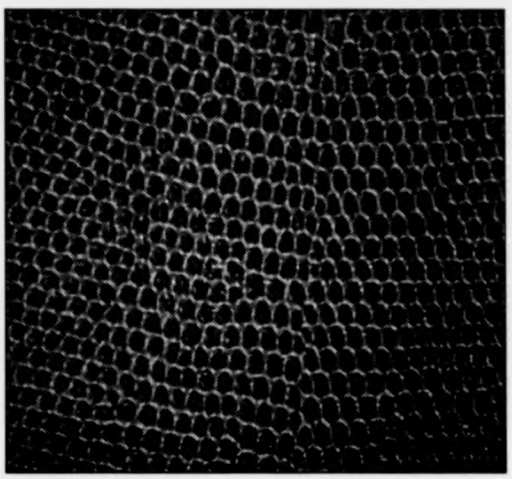
mean-square error, the scale for reptile skin is chosen at the MFT scale index 5, for burlap scale index 6 and for grass scale index 7. Figure 4.21 overlays the block size on the corresponding images. The scale at which an acceptable result is achieved obviously depends on the definition of 'acceptable': some applications will require a more accurate reconstruction than others. Nonetheless, for a given criterion of acceptability, the scale at which an acceptable result is achieved corresponds directly to the variability of the texture. To take this argument further, it would be necessary to study the quantisation of the parameters, which goes beyond the scope of the present work.

4.7 Summary

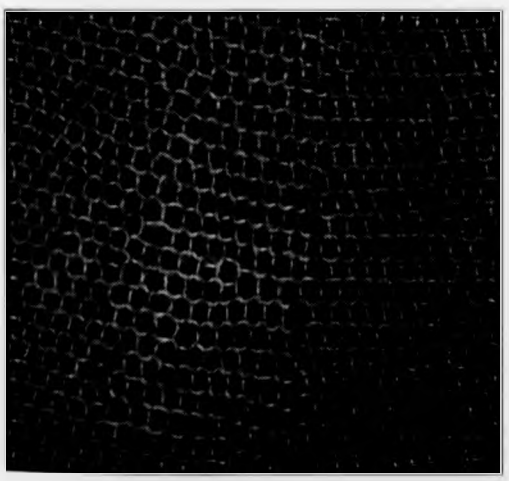
It has been shown that it is possible to model textures containing varying degrees of local structure by the use of affine transformation of a single prototype pattern. The accuracy of the reconstructions obtained using this technique demonstrates that the method captures the important features of the texture studied. Moreover, it is computationally efficient, requiring little more than one forward and one inverse FFT for each scale examined.



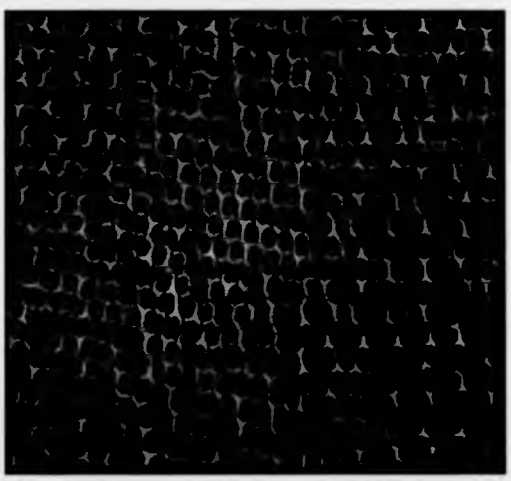
(a) Original Reptile



(b) Synthesis at level 6

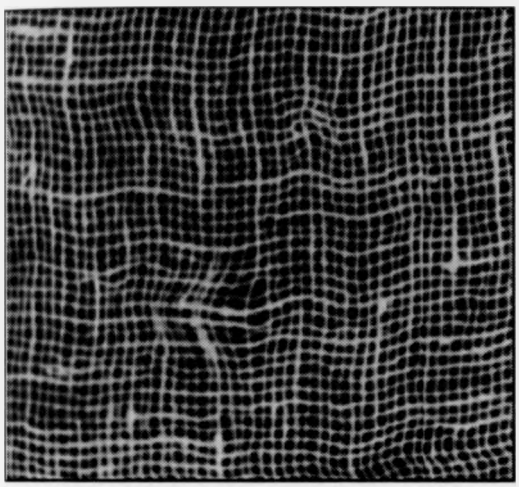


(c) Synthesis at level 5

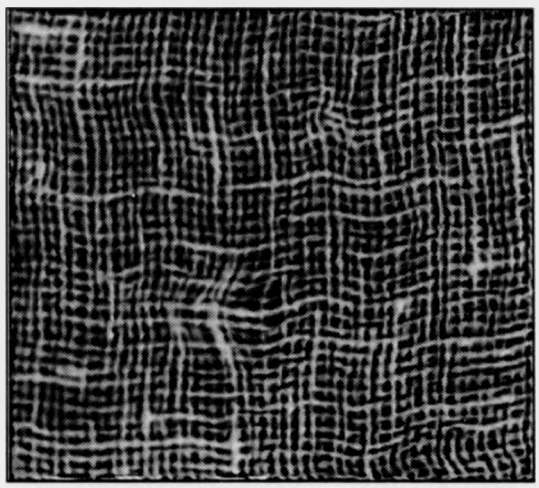


(d) Synthesis at level 4

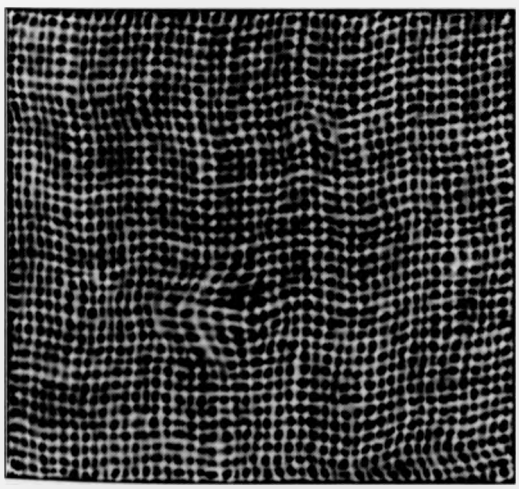
Figure 4.17: Reptile skin image synthesised at different levels.



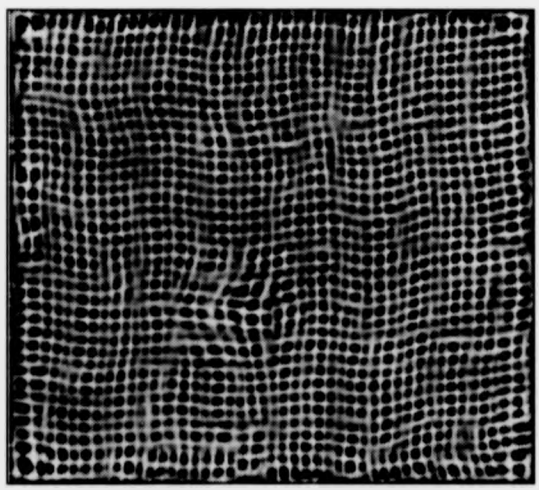
(a) Original Burlap



(b) Synthesis level 7

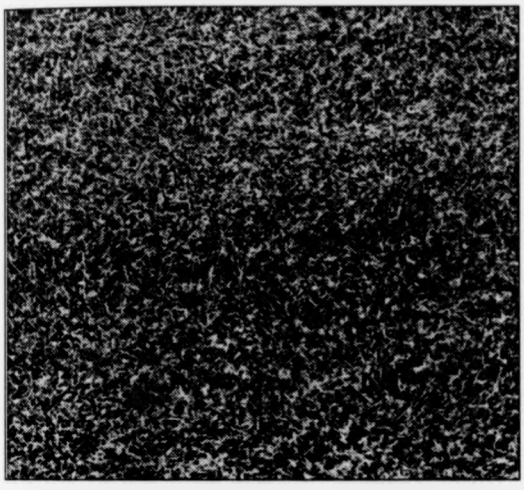


(c) Synthesis level 6

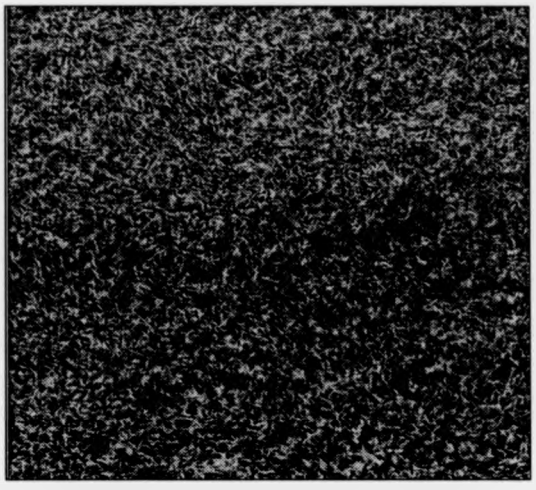


(d) Synthesis level 5

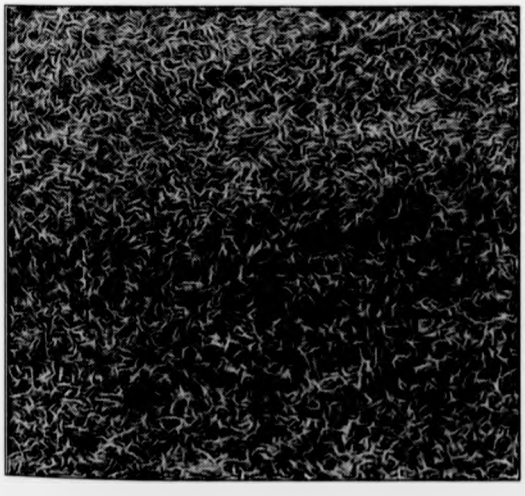
Figure 4.18: Burlap image synthesised at different levels.



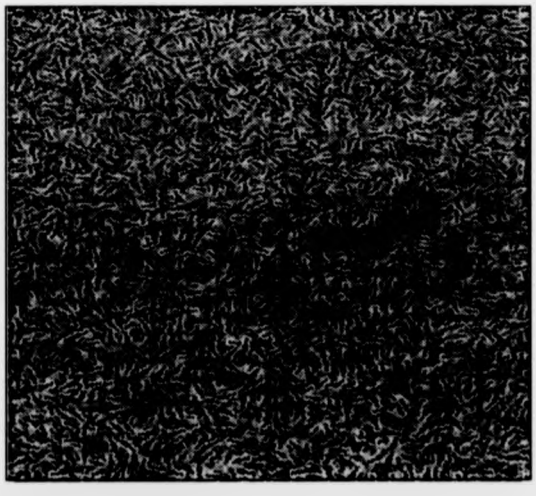
(a) Original Grass



(b) Synthesis level 7



(c) Synthesis level 6



(d) Synthesis level 5

Figure 4.19: Grass image synthesised at different levels.

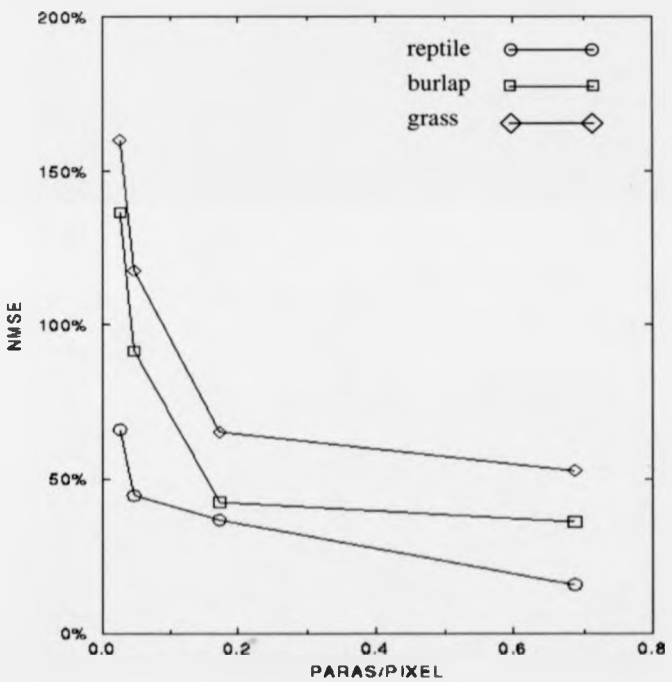


Figure 4.20: Normalised mean-square error as a function of the number of parameters used for synthesis.

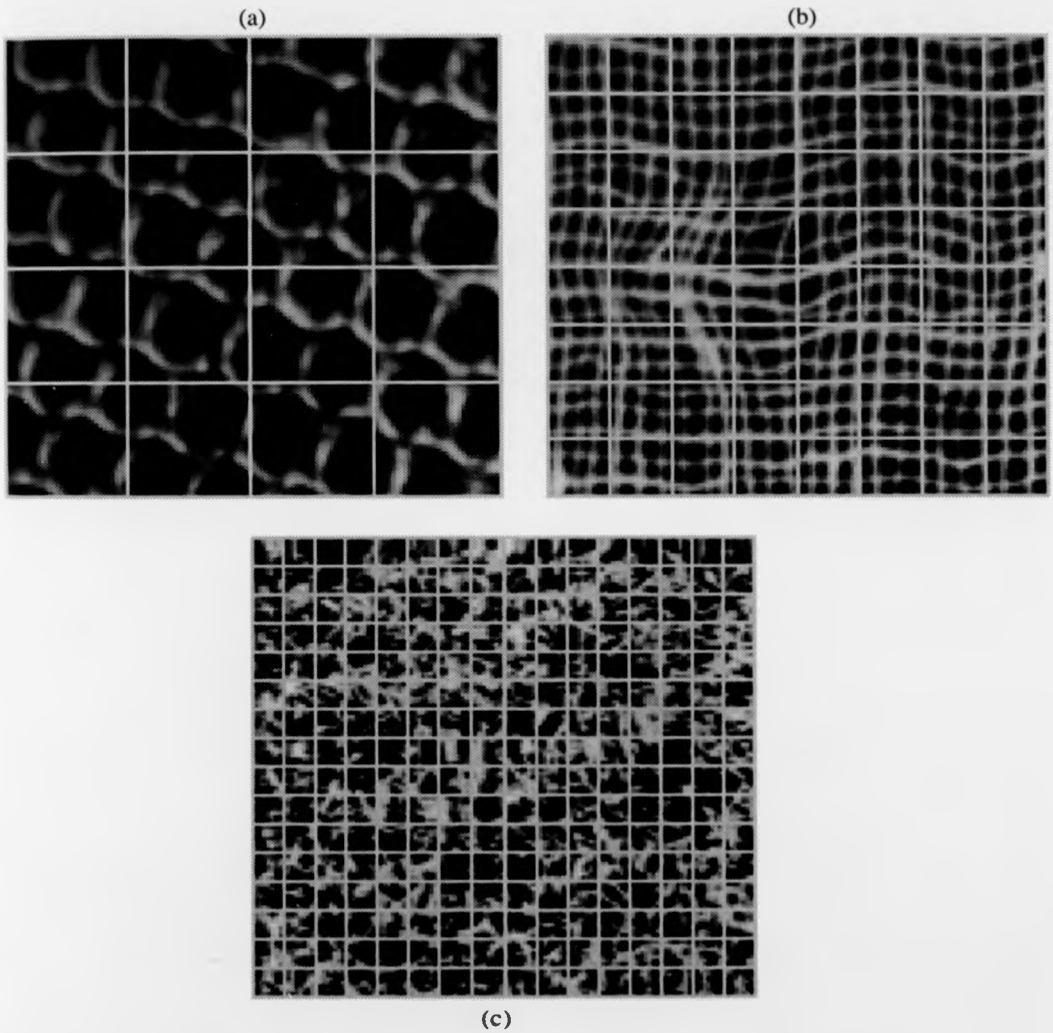


Figure 4.21: The choice of the prototype scale corresponds to different natural textural images: (a) reptile skin (b) burlap and (c) grass.

Chapter 5

Texture Segmentation

In the previous chapter, the main theme was to *synthesise* textured images, whereas the segmentation methods presented in this chapter are used to *analyse* the feature space for discrimination of textures. Segmentation issues are summarised and a texture segmentation algorithm based on the MFT is then presented. Experimental results are given to show the algorithm's performance. Finally, conclusions are drawn, with some comments and discussion of this algorithm.

5.1 Introduction

Segmenting an image into uniformly textured regions is a prerequisite for many computer vision and image understanding tasks [108]. Texture segmentation can be achieved either by extracting uniform texture regions or detecting textural discontinuities (texture edge detection) between image regions. It can be performed either in a supervised or an unsupervised mode, depending on whether a priori knowledge about the type and number of textures in an image is given or not.

5.1.1 Problems in Texture Segmentation

Textured images that occur in nature are varied in their spatial configurations. Texture synthesis based on the identification of an affine transform between a local region and the others in the image has been demonstrated. The results show that the synthesis has achieved a degree of success. It is now appropriate to consider how these ideas may be adapted to the discrimination of textures.

An intuitive idea in the analysis of the texture feature space is to employ the affine transform used in the synthesis. Surface texture and geometry affect the gray scale or colour of the image. Moreover, natural textures generally contain an inherent randomness or variability. This variation affects any local texture measure used in segmentation. Variability of the texture makes the correlation coefficient used to measure 'goodness of fit' in the synthesis unreliable in testing the validity of the affine model amongst local regions in an image. As a result, the correlation coefficient is not reliable in describing the texture's homogeneity and without it, there is no simple way to use the affine model directly for segmentation. On the other hand, the MFT coefficients clearly provide a rich set of features for discrimination. What is required is a way of identifying region boundaries without making unrealistic assumptions about the homogeneity of the texture within a given region.

Uncertainty and Segmentation

A fundamental difficulty in automatic segmentation centres around finding an adequate measurement of texture properties and the compatibility of simultaneously measuring texture features and their spatial position, both of which are essential in

segmentation. Uncertainty arises from the conflict between these two measurements [108]. In order to segment textured images into meaningful regions, a sufficiently large area averaging process is needed to reduce the fluctuations in texture properties. A consequence of the averaging is that the textured image becomes more homogeneous, but at the cost of blurring the boundaries. On the other hand, boundary detection methods having high spatial resolution are likely to generate spurious responses within the textured regions. To resolve this 'what is it' and 'where is it' problem, Wilson and Spann advocated the use of a multiresolution approach [108]. A useful characteristic of multiresolution approaches is region property invariance: those nodes in an estimate considered as interior to a region are given as the same class as their 'fathers' at lower resolution. This assumption of consistency of region properties across scales is a fundamental principle, which is related to the scale-space ideas introduced by Witkin [109] and refined by Koenderink [64].

5.1.2 Approaches to Texture Segmentation

Texture segmentation can be achieved either by detecting texture discontinuity between texture image regions or by extracting uniform texture regions. Therefore, segmentation methods are often categorised as boundary-based, region-based or as a hybrid of the two [33] [83].

Boundary-based Methods

The aim of this approach is to detect feature inhomogeneities. Wermser et al [98] proposed a method of estimating a 'texture gradient' by means of regression analysis performed on the outputs of a set of filters with different bandwidths and orientations.

This method can cope well with a variety of textures with different sizes of texture element. The method was reported as moderately successful, but few results were presented.

A hierarchical technique that has been considered for unsupervised texture segmentation was described by Granlund [48]. The texture boundary is detected by a general operator consisting of a set of orientation sensitive masks to allow the direction of an intensity change to be measured. The effect of the 1st-level transformation gives the magnitude and direction maximum change for the image texture, while the 2nd-level gives the boundary between textures. The segmentations produced by this method are relatively good when applied to textured images with distinctive directions, but discrimination is obviously limited by the use of a relatively small set of texture features.

Bovik, Clark and Geisler adopted a simple peak finding scheme and used 2-D Gabor functions as localised filters to represent images [9]. The Gabor amplitude and phase are used to locate the texture boundary. Their method can hardly be described as a general approach to the problem, however, as it tacitly assumes knowledge of the spectral properties of the textures to be segmented. Traditionally, this is done by statistical classification based on maximum likelihood or minimum distance techniques (eg [39]).

Region-based Methods

The basic premise of region-based methods is to seek feature homogeneities. Rosenfeld et al [80] used a method based on pyramid node linking. By linking regions (nodes) of one level with the most similar regions in adjacent levels, segmentation can be

achieved. Particularly, a combination of 'bottom-up' and 'top-down' linking was found effective.

Hybrid Approaches

Haddon and Boyce [50] combined region and boundary information in image segmentation. Their work uses two stages, based on gray level co-occurrence matrices. The initial pixel classification is defined by the distribution within a co-occurrence matrix and is used to segregate the region and boundary. This tentative segmentation is refined by relaxation labelling to take account of the assignments of nearby pixels. Local consistency is ensured by minimising the entropy of local information. One assumption made in this algorithm is that a co-occurrence matrix may be implemented as a feature space for those regions that satisfy Gaussian statistics; regions yield clusters on the main diagonal, and boundaries clusters off the main diagonal. The technique is reported as robust and performs well in poor imaging conditions.

Work reported by Spann and Wilson [91] uses a quadtree method using classification at the top level of the tree, followed by boundary refinement. A non-parametric clustering algorithm [105] is used to perform classification at the top level, yielding an initial boundary, followed by downward boundary estimation to refine the estimate. It is efficient and requires no a priori knowledge of the image regions. A generalisation of this work was applied to texture segmentation [104], [90]. The analysis method described in [91] is unable to cope with slow variation of structure caused by 3-D shape, which can result in large changes in the local spectrum across homogeneous regions. Such variation can be useful in estimating the 3-D structure [43] or in synthesis [97] and ought to be accommodated in any general texture model [55].

Summary

The requirements of the texture segmentation algorithm can be summarised as follows. The essential properties for texture segmentation are:

1. a set of texture features having good discrimination;
2. a segmentation algorithm having spatial constraints to process local information and reduce the uncertainty between class and position;
3. an estimate of texture features taking account the nonstationary nature of the textured regions;
4. no requirement for prior knowledge, so that segmentation is unsupervised and the process can be accomplished with minimal assumptions.

5.2 Texture Segmentation using the MFT

Given the effectiveness of the synthesis procedure described in Chapter 4, it is natural to base the segmentation on features derived from the MFT, which are the MFT coefficient magnitudes. As described in Chapter 3, the FPSS filters give optimal simultaneous concentration of signal energy in space and spatial frequency. In segmentation, the class resolution is determined by the bandwidth of the filters, a smaller bandwidth corresponding to a larger impulse response, giving higher class resolution and lower spatial resolution. Thus, the use of FPSS's leads to optimum class and spatial resolution in the segmentation.

There is good reason to think that the Fourier phase carries relatively little useful information for segmentation [35] [108], although its use in positioning the prototype

in synthesis is clearly essential. The Fourier phase depends on the relative position of the features. As an illustration, in figure 5.1, the figures in the left column show the single feature case; the right column shows a case of two features. The phase plots in (c) and (d) show no major difference between the two cases, whilst the magnitude plots in (e) and (f) show clearly that the presence of a pair of features in a given orientation has a large effect on the Fourier magnitude spectrum. Therefore, while the phase carries some information about the structure of the texture element, a significant amount is carried in the magnitude. Since averaging is necessary to reduce the effects of variations in the texture, it is convenient to use only magnitudes in the segmentation. Moreover, the energy spectrum has been shown to be a useful texture measurement [19] [91]. This suggests that the modulus of the MFT coefficient $|\hat{x}(\vec{\xi}, \vec{\omega}, \sigma)|$ is used as the feature set to be used in texture segmentation. Use of the Fourier energy spectrum is of course equivalent to using the autocorrelation of the signal.

5.2.1 Outline of Segmentation Algorithm

In analogy with the synthesis work presented in Chapter 4, the segmentation work makes use of the spatial localisation of the MFT to measure texture properties. In this process, both the boundary and region information are combined to place the texture boundary accurately. A pre-smoothing process is applied to the MFT magnitudes to reduce the texture fluctuations. Edge detection is performed by a simple pair of Sobel operators on the MFT magnitudes to give a texture boundary estimate. To reduce false edges and gaps, the estimated boundary is enhanced by making use of estimated boundary link probabilities. The gradient vectors are averaged to yield an

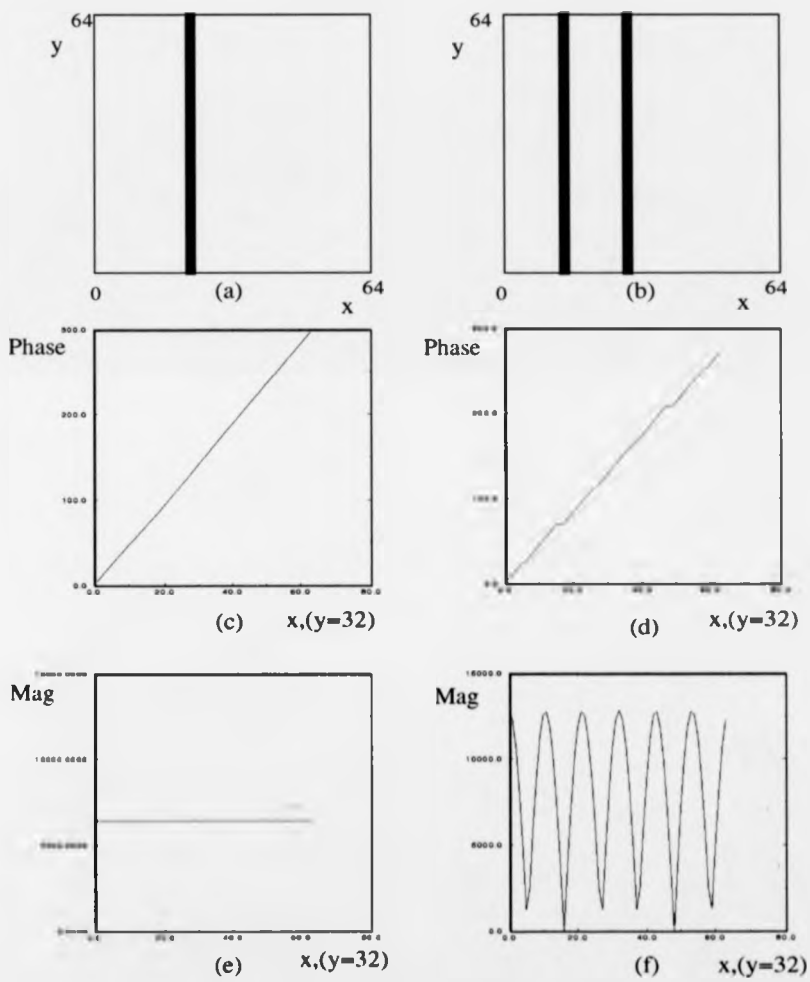


Figure 5.1: Fourier phase for various feature combinations: top figures are input images; middle figures are the corresponding Fourier phase plots in the horizontal direction; bottom figures show their amplitude spectra.

iteratively updated boundary estimate.

An iterative scheme is also adopted for local smoothing within regions. Similarly, the region properties are updated after each iteration by averaging with neighbours, based on region link probabilities.

The refined boundary estimate obtained at a given level from the averaging process is interpolated using a low pass kernel [101]. Following interpolation, the information is propagated down to the next resolution in a multiresolution framework, whereby both the needed boundary and region information are used successively until the finest spatial resolution is reached.

Interaction between the region and boundary processes is based on link probability estimates derived from the gradient estimation. An overview of the scheme is shown in figure 5.2.

5.3 Pre-smoothing Process

Local magnitude spectra are often highly variable. Under these circumstances, segmenting the image into regions is difficult. This can be alleviated by smoothing the MFT coefficient field prior to the classification process. The smoothing can be accomplished by simple local averaging of the MFT magnitude spectra, i.e. replacing the local magnitude spectrum at each MFT block by an average of neighbouring MFT blocks.

If the pre-smoothing operator S is given as

$$S = \frac{1}{9}\begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix} \tag{5.1}$$

then the smoothing operations are accomplished by the convolution operation among

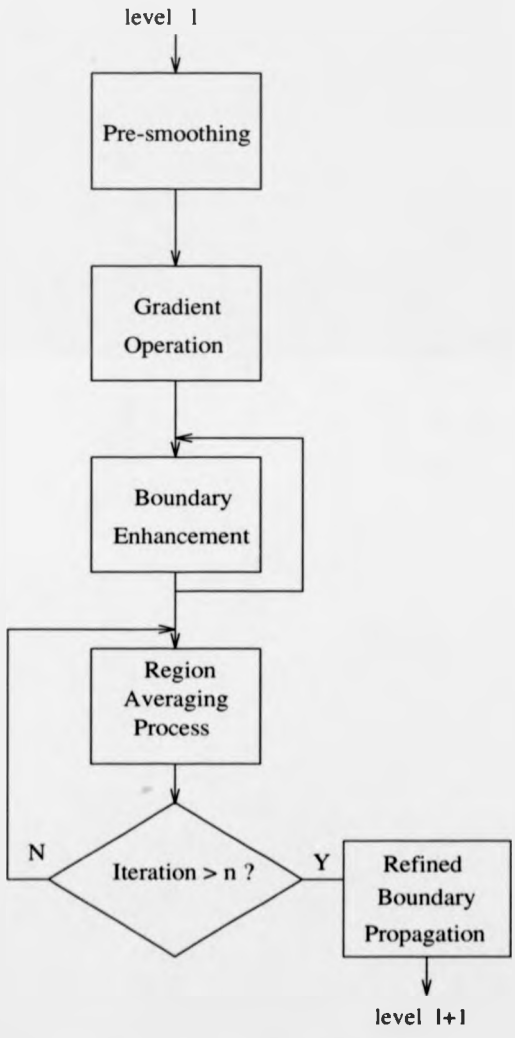


Figure 5.2: Outline of the texture segmentation at single level.

the blocks and within the blocks and implemented in a cartesian separable manner

$$\begin{aligned} a(\vec{\xi}_i, \vec{\omega}_j, l) &= S(\vec{\xi}_u) \overset{\vec{\xi}}{*} S(\vec{\omega}_v) \overset{\vec{\omega}}{*} |\hat{x}(\vec{\xi}_u, \vec{\omega}_v, l)| \\ &= \sum_{u,v} S(\vec{\xi}_i - \vec{\xi}_u) S(\vec{\omega}_j - \vec{\omega}_v) |\hat{x}(\vec{\xi}_u, \vec{\omega}_v, l)| \end{aligned} \tag{5.2}$$

where $|\hat{x}(\vec{\xi}_i, \vec{\omega}_j, l)|$ is the MFT coefficient magnitude at a given level l, $\overset{\vec{\xi}}{*}$ denotes spatial convolution and $\overset{\vec{\omega}}{*}$ denotes convolution in the frequency domain.

Within a given region, averaging reduces local fluctuations and hence the variability of the local spectrum, while preserving the mean value. Inevitably, averaging also blurs the borders of the region, but the boundary process will still extract the regions more or less correctly, although it will smooth out irregularities in their borders.

5.4 Gradient Vector Estimate

The texture boundary position between different regions can be considered as a location where texture properties change abruptly. This transition can be measured in terms of the gradient of a function x along r in a direction θ, that is,

$$\frac{\partial x}{\partial r} = \frac{\partial x}{\partial \xi}\frac{\partial \xi}{\partial r} + \frac{\partial x}{\partial \zeta}\frac{\partial \zeta}{\partial r} = x_\xi cos\theta + x_\zeta sin\theta \tag{5.3}$$

The magnitude of the gradient $\sqrt{x_\xi^2 + x_\zeta^2}$ corresponds to the amount of texture change across the boundary. In order to use gradient methods, two problems must be solved. First, a suitable definition of gradient must be used for the MFT. To solve this, a vector image has to be employed. Secondly, an appropriate representation of the gradient must be used.

5.4.1 Orientation Representation

To deal with the second point, the orientation estimate is to be used as contextual information for further processing (cf Section 5.6). In averaging operations, it is important to find a suitable orientation representation, which will prevent orientation sign ambiguity. Furthermore, unlike scalar boundaries, vector boundaries are not oriented (i.e. there is no ordering in R^n, $n > 1$). This leads to the requirement for an orientation representation which takes a line with orientation θ as the same as a line with orientation $\theta + \pi$. As noted by Granlund [48], a 'double angle' representation satisfies this requirement;

$$2\theta = 2(\theta + \pi) \bmod 2\pi \tag{5.4}$$

A gradient vector estimate $\vec{\theta}(\vec{\xi})$ in terms of the 'double angle' representation [63] is obtained as follows

$$\vec{\theta}_{\xi\zeta} = \begin{pmatrix} x_\xi^2 - x_\zeta^2 \\ 2x_\xi x_\zeta \end{pmatrix} = r_{\xi\zeta}^2 \begin{pmatrix} \cos 2\theta_{\xi\zeta} \\ \sin 2\theta_{\xi\zeta} \end{pmatrix} \tag{5.5}$$

where x_ξ and x_ζ are the gradient components, $r_{\xi\zeta}^2 = x_\xi^2 + x_\zeta^2$. This is the representation used in the segmentation algorithm.

5.4.2 Gradient Operation

The initial texture boundary detection is obtained by block-based gradient estimation. Using the pre-smoothed MFT local magnitude spectrum $a(\vec{\xi}_i, \vec{\omega}_j, l)$ with a block size of $N_l \times N_l$, the vertical and horizontal gradient components of texture boundary are given by

$$a^V(\vec{\xi}_\xi, \vec{\omega}_j, l) = a(\vec{\xi}_\xi, \vec{\omega}_j, l) \overset{\vec{\xi}}{*} S_V(\vec{\xi}) \tag{5.6}$$

and

$$a^H(\vec{\xi}_\xi, \vec{\omega}_j, l) = a(\vec{\xi}_\xi, \vec{\omega}_j, l) \overset{\vec{\xi}}{*} S_H(\vec{\xi}) \tag{5.7}$$

where $\overset{\vec{\xi}}{*}$ is convolution over spatial co-ordinates as in (5.2).

In this process, a Sobel operator [82] is employed, with a pair of kernels S_V and S_H defined as

$$S_V = \begin{pmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{pmatrix} \tag{5.8}$$

$$S_H = \begin{pmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{pmatrix} \tag{5.9}$$

Based on these two kernels, the gradients in vertical and horizontal directions can be obtained respectively, for each Fourier magnitude component $a(\vec{\omega}_i)$. These must then be combined to give an overall estimate of the spatial variation of the coefficients. Fortunately, there is a simple way to accomplish this, which uses the double angle representation. Since the representation of (5.5) can be applied independently to each MFT coefficient, all that is needed to obtain an overall estimate is to sum over all frequencies, giving

$$\vec{\theta}(\vec{\xi}, l) = \begin{pmatrix} a_{\mathrm{V}}^2(\vec{\xi}, l) - a_{\mathrm{H}}^2(\vec{\xi}, l) \\ 2a_{\mathrm{V}}(\vec{\xi}, l) a_{\mathrm{H}}(\vec{\xi}, l) \end{pmatrix} = r^2(\vec{\xi}) \begin{pmatrix} \cos 2\theta(\vec{\xi}, l) \\ \sin 2\theta(\vec{\xi}, l) \end{pmatrix} \tag{5.10}$$

where

$$a_{\mathrm{V}}^2(\vec{\xi}, l) = \sum_{i=0}^{N^2-1} a^V(\vec{\xi}, \vec{\omega}_i, l)\ a^V(\vec{\xi}, \vec{\omega}_i, l) \tag{5.11}$$

$$a_{\mathrm{H}}^2(\vec{\xi}, l) = \sum_{i=0}^{N^2-1} a^H(\vec{\xi}, \vec{\omega}_i, l)\ a^H(\vec{\xi}, \vec{\omega}_i, l) \tag{5.12}$$

$$a_{\mathrm{V}}(\vec{\xi}, l) a_{\mathrm{H}}(\vec{\xi}, l) = \sum_{i=0}^{N^2-1} a^V(\vec{\xi}, \vec{\omega}_i, l)\ a^H(\vec{\xi}, \vec{\omega}_i, l) \tag{5.13}$$

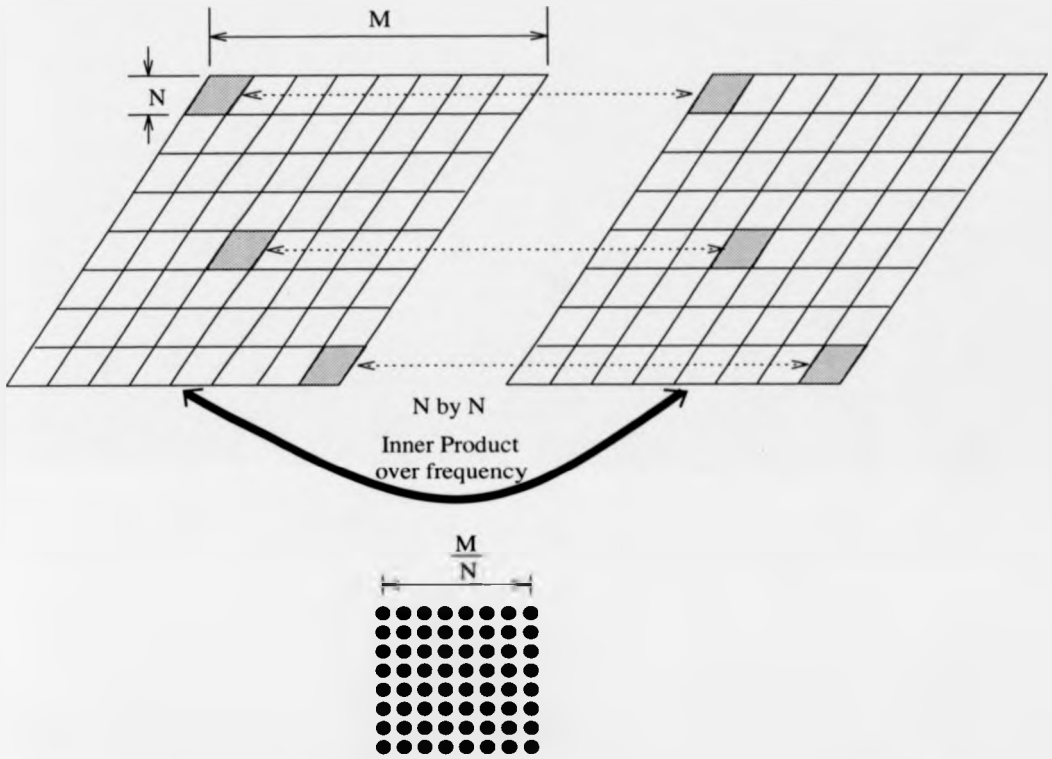


Figure 5.3: One level of the orientation pyramid is created corresponding to the MFT block-based gradient operation.

where N is the input MFT block size. As a consequence of the inner product with respect to the frequency co-ordinates, the orientation vector estimate is a subsampled image corresponding to the block size used. Each orientation vector corresponds to one MFT block of size $N \times N$. Figure 5.3 shows this mapping on one level.

This process is repeated over different scales of the MFT spectrum, creating an orientation pyramid (cf Bhalerao [7]), as shown in figure 5.13 for the test images.

5.5 Boundary Enhancement

5.5.1 Constrained Enhancement

The enhancement of the detected texture boundary makes use of a sigmoid function to give a notional 'link probability function P_b' defined as follows. Suppose for a given gradient vector estimate, $\vec{\theta}(\vec{\xi}, l)$, a weighting function f can be formed to measure the interaction between different vectors, given by

$$f(\vec{\xi}, \vec{\zeta}, l) = \frac{(\vec{\theta}(\vec{\xi}, l) + \vec{\theta}(\vec{\zeta}, l)) \cdot \vec{u}(\vec{\xi} - \vec{\zeta})}{\sigma_l} \tag{5.14}$$

where

$$\sigma_l^2 = \frac{1}{M^2} \sum_i \|\vec{\theta}(\vec{\xi}_i, l)\|^2 \tag{5.15}$$

and $\vec{u}(\vec{\xi} - \vec{\zeta})$ is the unit vector in the direction of the difference vector $(\vec{\zeta} - \vec{\xi})$, represented in double angle form. In effect, $f(\vec{\xi}, \vec{\zeta}, l)$ can be thought of as the interaction energy between pixels $\vec{\xi}$ and $\vec{\zeta}$, which is maximum when the link vector $(\vec{\xi} - \vec{\zeta})$ is collinear with the local orientation estimate. Averaging the orientation vectors at the two points ensures that the energy is symmetric, $f(\vec{\xi}, \vec{\zeta}, l) = f(\vec{\zeta}, \vec{\xi}, l)$. This form of energy function has been used by Yang and Wilson in a Hopfield neural network for edge detection [111]. Given this energy function, a simple way to estimate the link probability $P_b(\vec{\xi}, \vec{\zeta}, l)$ is via the logistic function, which is analogous to the Gibbs distribution approach used in Markov Random Field theory [45]. Specifically, if a neighbourhood system is in thermal equilibrium with its surroundings, then the probability of molecular configuration H_i is given by

$$P_b(i) = \frac{e^{-H_i}}{\sum_i e^{-H_i}} \tag{5.16}$$

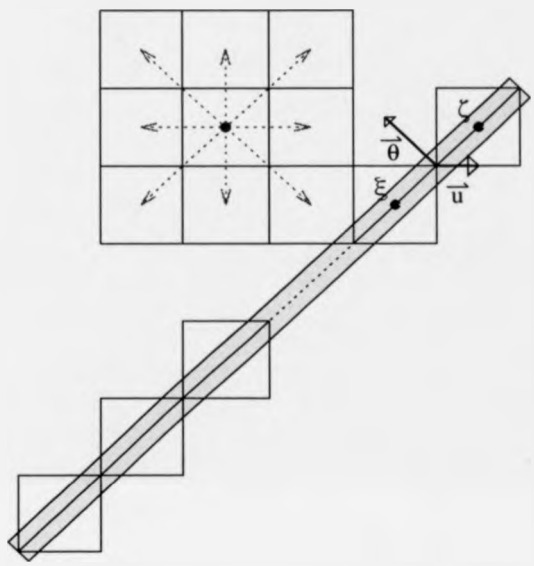


Figure 5.4: Boundary enhancement using 8 direct neighbour gradient vectors.

which is a Gibbs distribution with respect to the neighbourhood system. Analogously, the 'boundary link probability' function is written as

$$P_b(\vec{\xi},\vec{\zeta},l) = \frac{1}{1+\exp[-f(\vec{\xi},\vec{\zeta},l)]} \tag{5.17}$$

where 1 in the dominator represents a normalisation factor.

Boundary links are established between a given 'pixel' on some level l and its 8-neighbours, as shown in figure 5.4.

For each iteration n, the gradient vector $\vec{\theta}(\vec{\xi},l)$ is updated by weighted averaging, based on the link strengths

$$\vec{\theta}^n(\vec{\xi},l) = \frac{1}{\sum_{\vec{\zeta}\in N^8_{\vec{\xi}}} P_b(\vec{\xi},\vec{\zeta},l)} \sum_{\vec{\zeta}\in N^8_{\vec{\xi}}} P_b(\vec{\xi},\vec{\zeta},l)\vec{\theta}^{n-1}(\vec{\xi},l) \tag{5.18}$$

As a result of constrained enhancement, the link probability $P_b(\vec{\xi},\vec{\zeta},l)$ will be large if the boundary goes through blocks $\vec{\xi}$ and $\vec{\zeta}$ and low otherwise. The energy along the

boundary will tend to be redistributed along the direction of the boundary.

5.6 Region Averaging Process

5.6.1 Region Linking

The use of cooperative processing through the mechanism of probabilistic relaxation was first introduced by Rosenfeld et al. [86]. Probabilistic relaxation is an iterative approach for using neighbourhood properties and compatibilities to reduce local ambiguity. The scheme used here is similar to that used by Bhalerao [7] in two respects: it is an iterative scheme in which region properties are smoothed by neighbourhood operations; boundary and region processing is co-operative.

In this operation, the local links between blocks are iteratively updated. In analogy with the interaction coefficient in the boundary enhancement algorithm of section 5.5, for a given gradient $\vec{\theta}(\vec{\xi}, l)$ at $\vec{\xi}$, a region link probability of $\vec{\xi}$ and its four neighbours $\vec{\zeta} \in N_{\vec{\xi}}^4$, $P_R(\vec{\xi}, \vec{\zeta} | \vec{\theta}(\vec{\xi}, l))$, is approximated by

$$P_R(\vec{\xi}, \vec{\zeta} | \vec{\theta}(\vec{\xi}, l)) \propto \exp[-(f(\vec{\xi}, \vec{\zeta}, l)] \quad \forall \vec{\zeta} \in N_{\vec{\xi}}^4 \tag{5.19}$$

where

$$\sigma_l^2 = \frac{1}{M^2} \sum_i \| \vec{\theta}(\vec{\xi}_i, l) \|^2 \tag{5.20}$$

and $f(\vec{\xi}, \vec{\zeta}, l)$ is defined in equation 5.14. If the average vector in terms of double angle representation is given as

$$\begin{aligned} \vec{u}(\vec{\xi}, \vec{\zeta}, l) &= \vec{\theta}(\vec{\xi}, l) + \vec{\theta}(\vec{\zeta}, l) \\ &= \| \vec{u}(\vec{\xi}, \vec{\zeta}, l) \| \begin{pmatrix} cos2\theta(\vec{\xi}, \vec{\zeta}) \\ sin2\theta(\vec{\xi}, \vec{\zeta}) \end{pmatrix} \end{aligned} \tag{5.21}$$

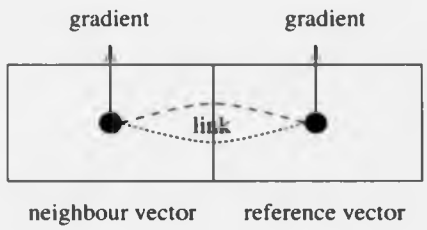


Figure 5.5: Case where adjacent pixels have high region and boundary link probabilities (cf. figure 5.15).

then

$$f(\vec{\xi},\vec{\zeta},l) = \| \vec{u}(\vec{\xi},\vec{\zeta},l) \| cos(\theta(\vec{\xi},\vec{\zeta}) - \phi(\vec{\xi},\vec{\zeta})) \tag{5.22}$$

and

$$\phi(\vec{\xi},\vec{\zeta}) = \mathrm{Arg}(\vec{\xi} - \vec{\zeta}) \tag{5.23}$$

In other words, blocks $\vec{\xi}$ and $\vec{\zeta}$ at level l are considered to be part of the same region unless the local gradient has large component parallel to the vector joining the pixels. Note that this is not the same as saying that there is a boundary link between them. The new local spectrum in the region averaging process at iteration n is obtained by weighting averaging

$$a^n(\vec{\xi},\vec{\omega},l) = \frac{\sum_{\vec{\zeta}\in N^4_{\vec{\xi}}} a^{n-1}(\vec{\zeta},\vec{\omega},l) P^n_R(\vec{\xi},\vec{\zeta}|\vec{\theta}(\vec{\xi},l))}{\sum_{\vec{\zeta}\in N^4_{\vec{\xi}}} P^n_R(\vec{\xi},\vec{\zeta}|\vec{\theta}(\vec{\xi},l))} \tag{5.24}$$

To compare the boundary and region links, figure 5.5 illustrates the case in which both links are high, because the boundary is parallel to the vector joining the two blocks.

After a number of iterations, the change in the boundary estimate should fall within some bound, terminating the process.

5.7 Propagation Process

5.7.1 Propagation of Estimates to Higher Resolutions

An orientation pyramid is created, based on the process described above, over a range of scales. Further enhancement of the boundary estimate is obtained on the basis of the multiresolution features, in a form of coarse-to-fine approach that assumes the invariance of region properties over a range of scales. In this multiresolution approach, evidence of the region properties at one resolution is passed to following resolutions. Based on the spatial consistency of the region, the accuracy of the boundary estimate can be improved (cf [108]).

Specifically, a boundary region is defined at the coarsest level by thresholding the region link probabilities. At successive levels, the blocks not classified as boundary region are assigned to the same class as their father. Otherwise, further estimation is performed to refine the candidate boundary by a factor of two using the MFT hierarchical structure. As a result, at the finest resolution a boundary at the full image size can be produced. The following algorithm is used for propagation.

stage1: Construct the region link probability on the largest scale of the orientation vector estimate.

stage2: Set link weighting function T_l by comparing with a given threshold value

$$T_l(\vec{\xi},\vec{\zeta}) = \begin{cases} 1 & \text{if } P_R(\vec{\xi},\vec{\zeta}|\vec{\theta}(\vec{\xi},l)) > \text{THRESHOLD} \\ 0 & \text{otherwise} \end{cases} \tag{5.25}$$

where the THRESHOLD is given as 0.75 which assumes that 75% of the links are 'on'.

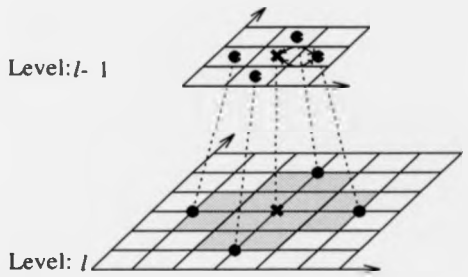


Figure 5.6: Consistency check among 4 neighbour gradient vectors against their father node.

stage3: Construct the region link probability at one level down only when their father's link weighting functions with its 4 neighbours are not all mutual 'on' shown as figure 5.6. Those nodes whose father's link weighting functions with their 4 neighbours are all mutual 'on' are forced to be 'on' too. Hence, the more reliable estimate from higher levels can be propagated down to lower levels.

$$T_l(2\vec{\xi}+\vec{m}, 2\vec{\zeta}+\vec{n}) = \begin{cases} 1 & \text{if}(T_{l-1}(\vec{\xi},\vec{\zeta}) = 1) \\ 0 & \text{otherwise} \end{cases} \tag{5.26}$$

where $\vec{m}$ and $\vec{n}$ represent the relative addresses of the children on level l of a node on level $l-1$.

stage4: Refine the boundary estimate by further checking the nodes where their fathers' link weighting function is not on. Because the strength of region links along the boundary is less than that in the homogeneous area, further iterations are performed on these links. Meanwhile, boundary enhancement is also performed only over those blocks where the region links are not on to avoid the emergence of spurious boundary nodes.

stage5: Generate a new version of the MFT magnitude spectrum as in (5.24).

As a result of one level of this process, the regions are averaged and the boundary enhanced.

5.7.2 Boundary Propagation

In the multiresolution approach, features detected at higher levels are generally more reliable than those from lower levels, as the bigger window size reduces fluctuations. Thus, interpolation of the boundary estimated from higher levels may be used to improve the accuracy of the gradient estimate at following levels.

This principle of propagating gradient estimates from low to high resolution was first established by Clippingdale [20] and has been used by Bhalerao [7] and Yang [111]. The gradient estimate resulting from a higher level is interpolated and combined with the gradient estimate from the current level as follows

$$\vec{\theta}(\vec{\xi},l) = \beta(l) \sum_{m=-K}^{K} \sum_{n=-K}^{K} w(m,n)\vec{\theta}(\xi_1/2+m, \xi_2/2+n, l-1) + (1-\beta(l))\underline{\vec{\theta}}(\xi_1,\xi_2,l) \tag{5.27}$$

where $\vec{\xi} = (\xi_1, \xi_2)^T$ is the spatial co-ordinate, $w(m,n)$ is the low pass kernel used as the interpolation filter [101], and $\underline{\vec{\theta}}(\vec{\xi},l)$ is the gradient estimate before propagation. A 4×4 version of this lowpass filter is employed in this work, so that the computational cost is kept at a minimum. The coefficient $\beta(l)$ is the variability ratio of the gradient estimate, which controls the amount of information propagated down to the lower level and is based on the measured gradient variances at the levels,

$$\beta(l) = \frac{\sigma_{l-1}^2}{\sigma_l^2 + \sigma_{l-1}^2} \tag{5.28}$$

where σ_l^2 is the average value of the magnitude square of the gradient estimate (cf 5.20). If the value of $\beta(l)$ is large, which indicates larger variation of the gradient

estimate obtained at the lower level, then more information from the higher level is needed. This will typically occur when the block size of the MFT is smaller than the texture element size.

5.8 Presentation and Description of Results

Various experiments have been performed on both synthetic and natural images of size 256×256 pixels to test the efficacy of the method. Note that in these experiments extrapolation of the blocks on the image boundary was used to minimise edge artifacts.

The textured images are synthesised based on the discussion of Chapter 3. The texture boundary is formed using a linear recursion equation. For a sample of white noise $u(n)$, the boundary position $b(n)$ on line n is defined as

$$b(n) = b(n-1) + w(n) \tag{5.29}$$

where $w(n)$ is the increment, defined by

$$w(n) = 0.9\ w(n-1) + 0.43\ u(n) \tag{5.30}$$

As a result of the linear recursion, the current position on the boundary depends on the last position of the boundary so that the boundary line is generated smoothly.

The experiments are performed over 5 scales and 4 texture pairs with results of $1, 2, 4$ and 8 iterations. The 4 texture pairs are depicted in figure 5.7, 5.8, 5.9 and 5.10.

Figure 5.7: Synthetic structured texture pair.

Figure 5.8: Synthetic random texture pair.

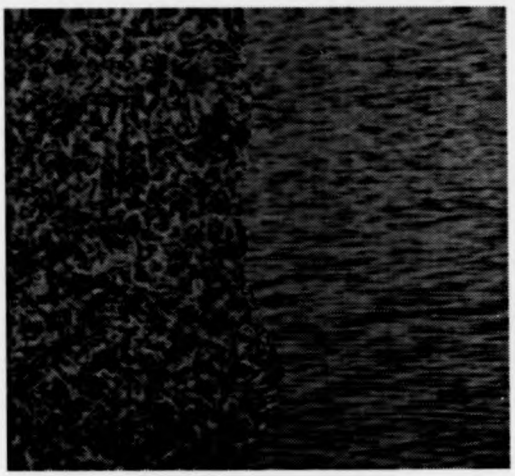


Figure 5.9: Natural random texture pair.

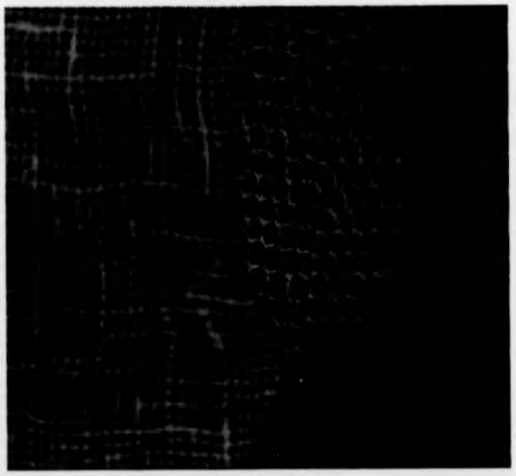


Figure 5.10: Natural structured texture pair.

Figure 5.11: Synthesised colour reference map for orientation.

5.8.1 Gradient Pyramid

The orientation is represented in double angle such that one colour corresponds to an angle θ and its complement to $\theta + \frac{\pi}{2}$. The colour reference map is displayed in figure 5.11.

The gradient estimate of the texture boundary consists of two components, the magnitude and the orientation. Figure 5.12 shows the orientation estimates of four test images over MFT levels $3, 4, 5$ and 6, with MFT block sizes of 16×16, 32×32, 64×64 and 128×128. As a consequence of the random fluctuations in the textures, there exist spurious edges in the orientation estimates. These spurious edges are reduced gradually after applying the iterative estimation process, while the boundary magnitude is preserved (cf section 5.8.4). The magnitudes of the preliminary estimate of the texture boundary are shown in figure 5.13.

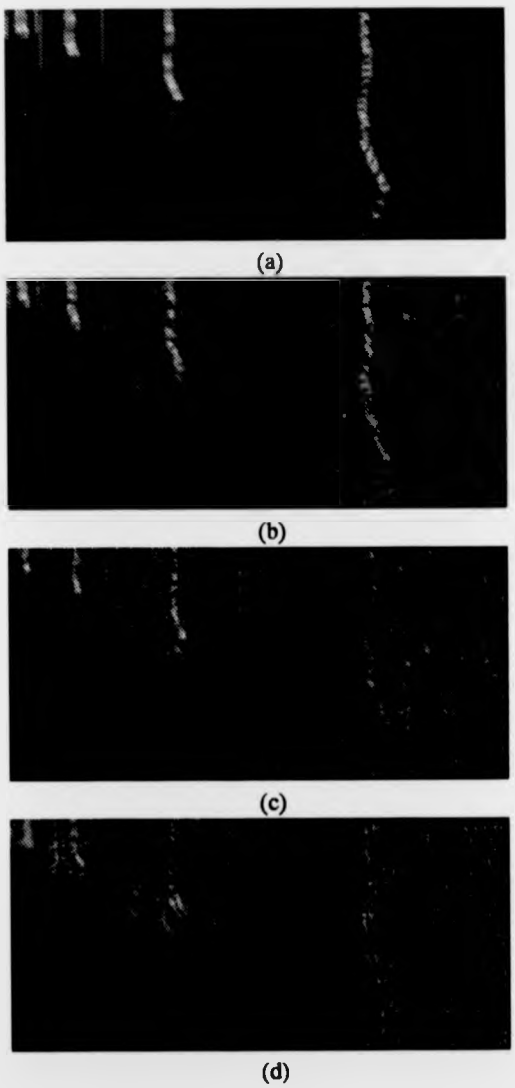


Figure 5.12: Orientation pyramids of testing texture pairs. (a) Synthetic structured texture. (b) Synthetic random texture. (c) Natural random texture grass water. (d) Natural structured texture burlap reptile skin.

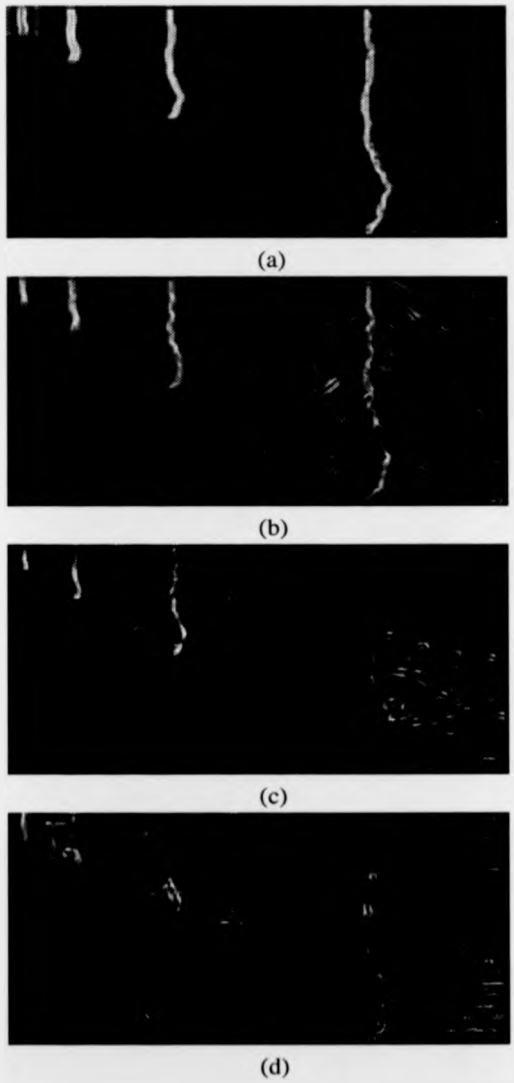


Figure 5.13: Magnitudes of orientation estimates correspond to those in figure 5.12. (a) Synthetic structured texture. (b) Synthetic random texture. (c) Natural random texture grass water. (d) Natural structured texture burlap reptile skin.

5.8.2 Boundary Link Probability

Figure 5.14 displays the boundary link probability of the grass-water texture pairs at the MFT block scale of 16 × 16 and scale of 32 × 32. (a) shows the boundary link strength on the first iteration; the enhanced boundary link after three iterations is shown in (b); (c) and (d) show the results with different scales. The boundary link probabilities depicted in figure 5.14 show that the link strength between two points along the boundary has been enhanced if their orientations are similar, but otherwise the link strength has been reduced. In particular, the boundary breaks which occur at the image edge have been reconnected.

5.8.3 Region Link Probability

The region blocks are averaged using the region link probability as the weighting function during the iteration process. Figure 5.15 shows the region link probabilities of the grass-water texture pair at the MFT block size of 16 × 16. (a) depicts the region link strength of the first relaxation process; two ambiguous regions of the right hand homogeneous region are joined after eight iterations as shown in (b). Thus, the iterative smoothing removes much of variation in the original data, while preserving the boundary.

5.8.4 Segmentation Results

Since the segmentation results are produced from one level and propagated down to the next level, it is useful to show the generating order of the results, as in figure 5.16. The results of the iterative estimation of the synthetic random texture of figure 5.8 are shown in figure 5.17. The results of the iterative estimation of the natural texture

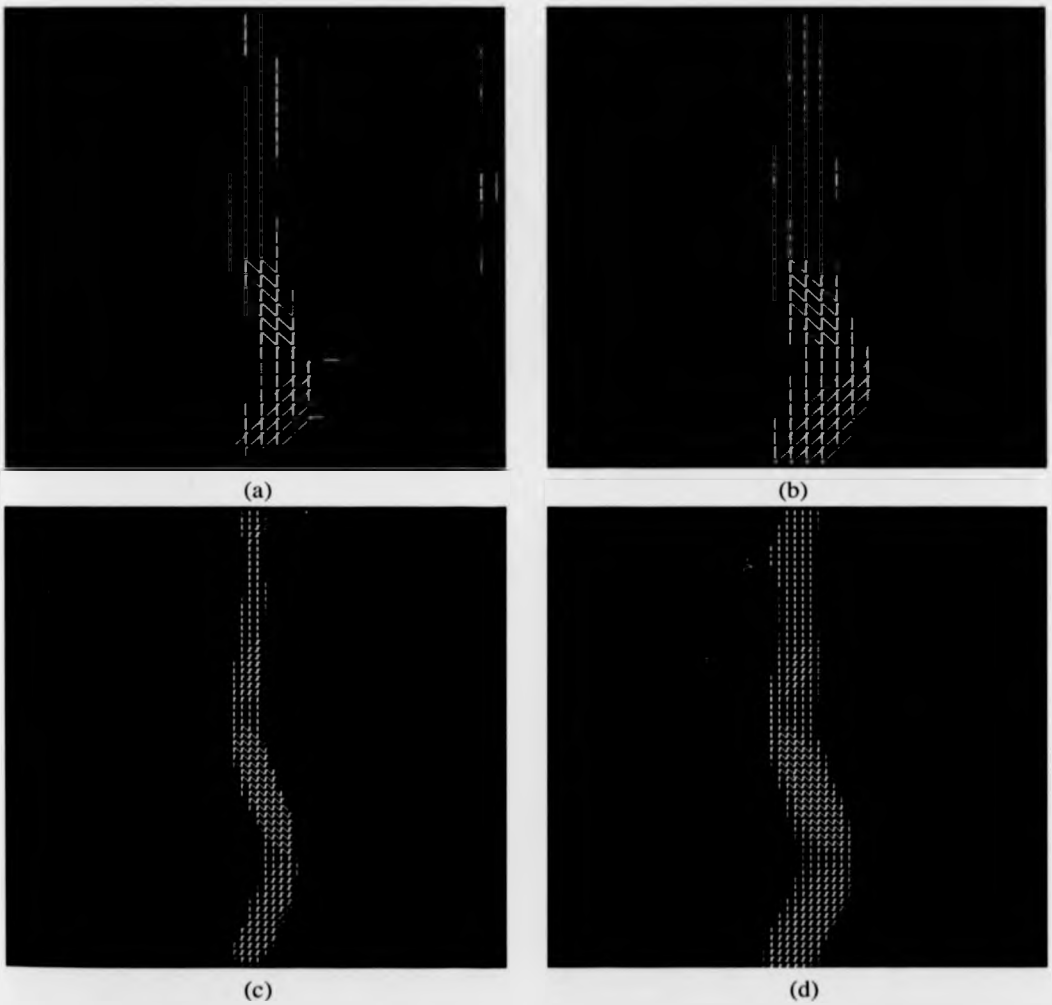
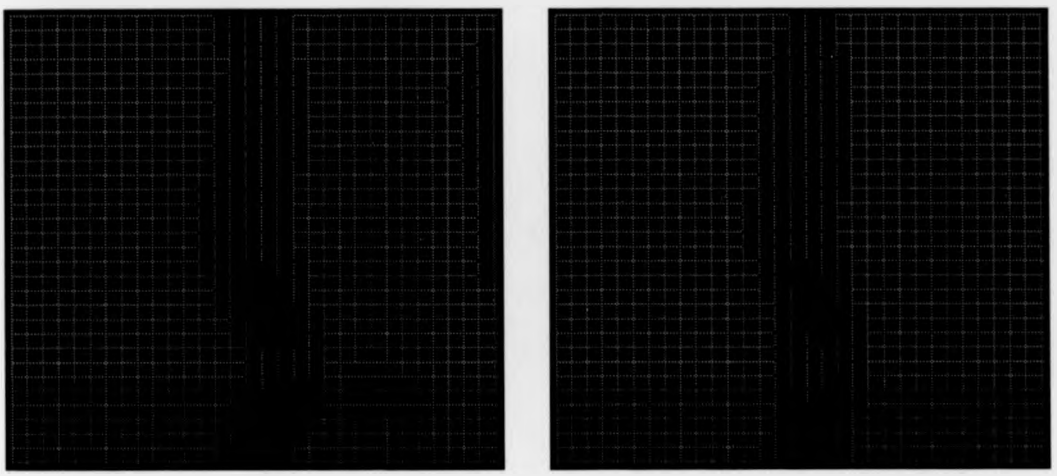


Figure 5.14: Boundary estimate is enhanced via boundary link probability iteratively. (a) Boundary links at MFT block scale of size 16 × 16. (b) Boundary links after 3 iterations. (c) Boundary links at MFT block size 8 × 8. (d) Boundary links after 3 iterations.



(a) (b)

Figure 5.15: Region connectivity is described via region link probability iteratively.

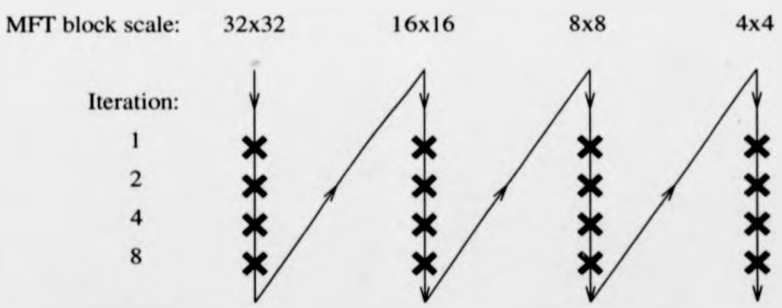


Figure 5.16: The generating order of the segmentation results illustrated in the following figures (cf 5.17, 5.18 and 5.19).

of figures 5.9 and 5.10 are shown in figures 5.18 and 5.19 respectively, where the spurious edges have been removed. For the burlap-reptile texture pair in figure 5.19, the detected boundary reveals two high magnitude boundaries at high resolutions. For the 128×128 boundary estimate, the MFT block size of 4×4 is much smaller than the size of texture element of both burlap and reptile. The result is that the MFT at this scale acts primarily as a luminance edge detector, so that the action of the Sobel operators on this detected edge gives a double response. Otherwise, the results show that the boundary can be located quite readily.

5.9 Conclusions

A segmentation algorithm based on the MFT has been presented in this chapter. The unification of both boundary and region information is employed in the algorithm. A block-based Sobel operator is first implemented to yield a boundary gradient estimate. The use of the double angle representation to indicate the orientation is an effective method to detect boundaries in vector fields. The boundaries obtained here show some uncertainty because they are derived from area properties. If the MFT block size is smaller than the size of the texture element, spurious edges are obtained in the initial estimate. Hence, cooperative processing of the boundary information is employed. This uses estimated link probabilities to enhance the initial boundary estimate. Also, boundary vectors that are generated at the coarse resolutions are propagated through to the next level to constrain the estimate of the texture boundary. This has the advantage of giving a sharp and accurate boundary, while using only local operations, resulting in computational savings. Another advantage of using iterative linking is that it does not require initial parameter settings or prior knowledge of the number

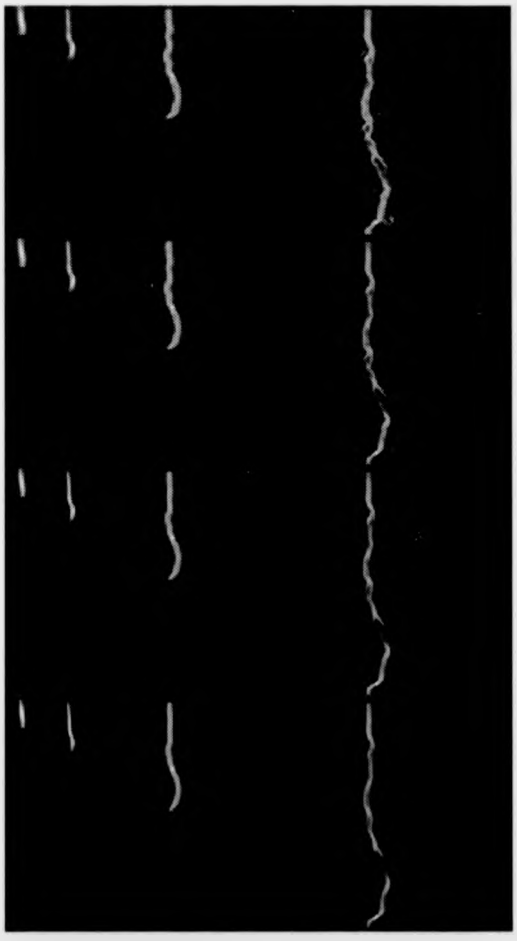


Figure 5.17: Iteration results of orientation pyramid of synthetic random texture.

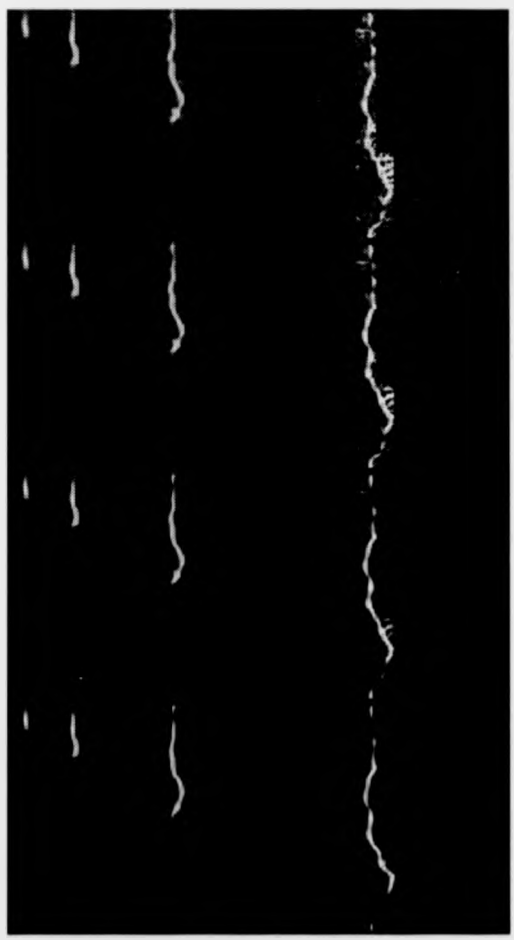


Figure 5.18: Iteration results of orientation pyramid of grass-water texture pair.

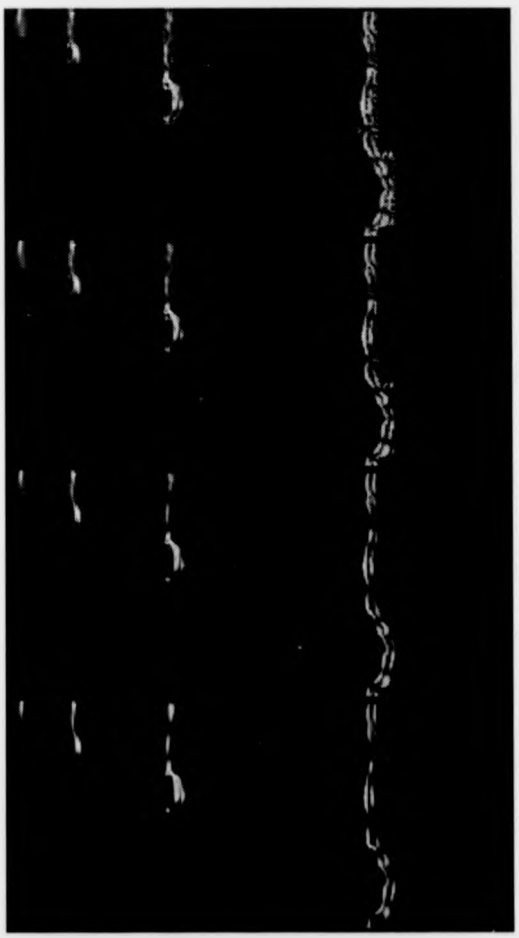


Figure 5.19: Iteration results of orientation pyramid of burlap-reptile texture pair.

or size of the regions.

By basing all the processing on local operations, no assumption of uniformity of region properties is required, unlike the work of Spann and Wilson [91], [108], for example.

Chapter 6

Conclusion

This work has been concerned with texture synthesis and analysis using the MFT. It consists of two parts: texture synthesis and texture segmentation.

6.1 Thesis Summary

Chapter 1 introduced the problems of texture analysis and general properties required for description of texture, in connection with the human vision system. Previous work on texture analysis was discussed. The issues of uncertainty and use of scale in texture analysis were brought up. The motivation for use of a multiresolution approach was discussed. An outline of the thesis was given.

In Chapter 2, the Fourier related approaches to texture were discussed further. The requirements of an adequate texture descriptor, such as linearity, invertibility, symmetry and locality were considered, in order to find a suitable texture analysis method. The Fourier transform based approaches lead to a rich mathematical structure associated with the complex exponential functions, which form a complete orthogonal set. It has useful properties in that symmetries under dilation, translation and rotation are preserved within the transform. In respect of locality, however, the

Fourier transform has to be enhanced in order to become a general texture analysis tool. An obvious enhancement would be to incorporate scale and locality properties. To combine spatial and spatial frequency methods, windowed Fourier transform methods are an obvious approach. By employing the Gabor function as a window function, the Fourier transform becomes a localised transform. However, the Gabor signal representation was shown to be limited by its fixed resolution. A solution to this problem is to use multiresolution methods.

A number of interesting features of the multiresolution pyramid and the wavelet transform were summarised. The multiresolution pyramid and the wavelet transform can both well describe localised image features over different scales. The wavelet transform is an invertible representation which is adapted to some of the natural symmetries of the vision problem. Both methods also have the potential for dealing with the complexity problem by systematic variation of the scale at which the analysis task is performed, and provide a rigorous basis for multiresolution analysis. However, the multiresolution pyramid representation is isotropic: it has no oriented features. The wavelet transform is computationally fast and requires less storage than the multiresolution pyramid, but has limitations in terms of symmetry. Because the wavelet transform is based on a 3 parameter subgroup of the 2-D affine transform, it does not include the rotation and shearing that are essential for texture processing. For texture analysis, a more general transform is required.

To overcome these limitations, a general multiscale windowed Fourier transform - the multiresolution Fourier transform (MFT) - is used. The MFT has the ability to represent arbitrary signals of compact support over a range of resolutions in an interference-free manner. In this hierarchical representation, the MFT can trade off

the resolution in spatial and spatial frequency domains by increasing the resolution in the spatial domain and decreasing the resolution in the frequency domain in a systematic way.

In Chapter 3, the MFT definition and properties were given. The MFT can be interpreted both as the conventional Fourier transform and as a linear filter bank scheme. It follows that there are two corresponding implementations: the spatial implementation and the frequency implementation. The spatial implementation uses an index-limited window function and the frequency implementation uses a band-limited window function.

The usefulness of the MFT in both the spatial and frequency implementations is constrained by the choice of the window function and sampling rate. The primary purpose of the window in the windowed Fourier transform is to limit the extent of the input signal to be transformed so that the spectral characteristics are reasonably 'stationary' over the duration of the window. To resolve the uncertainty problem, the window function used in the MFT is a band-limited Finite Prolate Spheroidal Sequence (FPSS). The FPSS is used as a windowing function with maximal simultaneous energy concentration in both spatial and frequency domains. An oversampling scheme is used in the implementation. This results in an overlap between adjacent windows, which is helpful in reducing artifacts in the synthesis.

The implementation of both the forward and inverse transforms was discussed. It was shown that both are efficiently implemented using the FFT algorithm. The MFT spatial implementation, which makes use of the index-limited window function, introduces artifacts in the frequency domain when the Fourier transform is taken. This is undesirable for applications in texture analysis, in which spectral properties

are significant.

Some preliminary results were presented to demonstrate the MFT properties. The emphasis was on the MFT features over a range of scales and the description in terms of the localised orientation and texture granularity.

A new frequency domain approach to the description of texture is described in Chapter 4. Both statistical and structural aspects of the problem are combined within this framework. The method works by creating an approximation to the test data from a suitable prototype, based on the identification of the affine co-ordinate transformation which gives the best match between prototype and test patch. By identifying the affine transform between local patches of the image, the image can be synthesised from a prototype texture element warped using the inverse affine transform.

The identification of the affine transform between local patches of the image consists of three main components:

1. Determination of the linear co-ordinate transformation which gives the best fit between the magnitude spectra.

2. Linear transformation of the prototype spectrum to give an approximation to that of the texture element.

3. Correlation of the texture element and transformed prototype by multiplication of the respective spectra and inverse Fourier transformation, giving a 'goodness of fit' measure and an estimate of the spatial translation using the peak of the correlation.

One pair of representative centroids is estimated from each MFT magnitude spectrum to form the affine transform parameters. The estimate is accomplished by finding centroids with the minimum sum of variances, on the basis of the angular distribution of spectral energy. Selection of minimum variance classes and centroids in this way ensures that the classification is robust under linear transformation - an essential feature of any method being used to identify the linear transform relating the two spectra.

Both the spectrum magnitude, used in determination of linear component of affine transform and the phase, used in the determination of the translation, are utilised in this method. In this respect, this method is different from earlier methods, which usually ignore the phase information [104]. In this way, the ability to deal with textured surfaces with geometric warping due to 3-D projection is significantly improved.

The accuracy of the reconstructions obtained using this technique demonstrated that the method captures the important features of the texture and has successfully extended the work of Spann and Wilson [108]. Two measures of performance were defined: the number of parameters per pixel being used in estimating affine transform and the SNR between the original and the synthesised texture obtained at each MFT level; these may be used to analyse the scale of the texture. Moreover, the process is computationally efficient, requiring little more than one forward and one inverse FFT for each scale examined.

Chapter 5 presented a scheme for texture segmentation, based on a cooperative algorithm within the MFT framework. A number of existing texture segmentation works were reviewed, which are classified into boundary-based methods, region-based methods and methods that are a hybrid of the two. As a result of the review, the

essential properties for texture segmentations were found to be:

1. A set of texture features having good discriminatory abilities.

2. A segmentation algorithm having spatial constraints so that it is able to process local information and can reduce the uncertainty between class properties and position information.

3. An estimate of texture features which takes into account the nonstationary characteristics of typical natural textures.

4. No requirement for prior knowledge, so that segmentation is unsupervised and the process can be accomplished with minimal assumptions.

In analogy with the synthesis work presented in Chapter 4, the texture segmentation work makes use of the spatial localisation of the MFT to measure texture properties. The features used are the MFT coefficient magnitudes. As has been shown by Spann [90] that the Fourier phase carries less useful information for segmentation, although its use in positioning the prototype in the synthesis method of Chapter 4 was important. Moreover, the segmentation process requires averaging, which can only be effective if the linear phase variation is removed. The simplest way to achieve this is by removing the phase altogether.

There are five main steps in the segmentation for each MFT level:

1. Pre-smoothing to reduce texture fluctuations.

2. Edge detection to give an initial texture boundary estimate.

3. Boundary enhancement to reconnect the gaps in the estimated boundaries.

4. Region averaging to refine the estimated boundaries.

5. Propagation across scale to transmit the more reliable information from the larger scales down to the lower scales, where spatial resolution is higher.

In this process, both the boundary and region information have been combined to place texture boundaries accurately. Edge detection is performed by a simple Sobel derivative operation on the MFT magnitudes of the image to give an initial texture boundary estimate. This process has shown how boundaries in vector fields can be detected by making use of the double angle representation to indicate the orientation.

The estimated boundary reveals some uncertainty because of the area properties used in the boundary detection process. In particular, if the MFT block size is smaller than the size of the texture element, spurious edges are obtained in the initial estimate. A method that enhances the estimated boundaries by reducing the false edges and gaps has been developed. It makes use of a logistic function to give a notional 'link probability function', so that the boundary links are established between a given 'pixel' and its 8-neighbours on some level. By applying the averaging process, each gradient vector is replaced iteratively by its 8 neighbours weighted with boundary links, resulting in an enhanced boundary estimate.

An iterative scheme was also adopted for local interaction within regions. The region properties are updated after each iteration by averaging with neighbours based on the estimated region link probability. Both boundary link probabilities and region link probabilities introduce dependence on neighbouring data, allowing the boundary to be refined successively to achieve segmentation.

Similarly, there are two different types of information propagated between levels

to improve estimates at small scales. The region link functions and refined boundary obtained from the lower resolutions are propagated down to the higher resolutions. The propagated link functions restrict the process at the next level to the area where the region links propagated from the last level are not 'on'. Meanwhile, the estimated boundaries obtained at the higher level are propagated by interpolation to reduce noise in the gradient estimate. Although the principle of the propagating estimate from low to high resolution in segmentation has been used by Clippingdale [20], Bhalerao [7] and Yang [111], it has been shown how to embed this algorithm within a MFT framework. Encouraging results have been obtained, which illustrate the efficacy of the MFT as a representation and the performance of the segmentation algorithm.

Although methods based on Markov Random Fields (MRF's) have been applied successfully to texture description and segmentation, they are unlikely to be able to describe the combination of structure and variation exhibited, for example, by the reptile skin. Since these methods are pixel-based, for textured images showing large scale structure, MRF's methods are inappropriate for analysis. The advantage of the new method is its flexibility: the scale of the MFT blocks can be chosen to synthesise any desired structure, ranging from random textures and irregular textures to regular textures. Although nonlinear MRF models can be used to describe textures having large scale structure, estimation of the conditional densities required by these models is a computationally expensive task, requiring large data sets to achieve satisfactory results.

There are a number of interesting features of the segmentation process. The boundary link probability and region link probability are incorporated cooperatively in the

iterative averaging process. In this way, the segmentation process incorporates the advantages of both, in terms of the computational efficiency and in giving a sharp but accurate boundary. Furthermore, no assumption about the uniformity of region properties is required in this algorithm, unlike the work of Spann and Wilson [108], nor about the spectral characteristics of the texture, as in [9].

6.2 Limitations and Further Work

The synthesis and the segmentation results obtained from the experiments have demonstrated the effectiveness of the methods. However, there are a number of issues that need to be considered further.

1. Model Extension: A process which combines the synthesis and segmentation algorithms could be developed, provided that some sophisticated tessellation method is applied in the areas along the boundary. That is, once the texture boundary is obtained, the results of the region link probabilities and boundary link probability lead to the discrimination of texture regions automatically. For example, those regions where the region link probabilities are 'on' can be discriminated as some type of texture region. Within these regions, the texture synthesis process proposed in this work can be applied. In order to do this, the question of scale selection is important, but it might be naturally resolved by dealing with regions at the scale at which they appear. In this way, the work described could be extended to a full analysis/synthesis system.

2. Linear Phase: Fourier phase is important for signal representation [74]; many of the features of a signal are preserved if only the phase is retained. However, the

segmentation algorithm proposed uses iterative averaging, in which the variations of Fourier phase tend to reduce the magnitude during averaging process. It may, however, be possible to remove the linear phase term from the MFT blocks, providing a suitable 'normalised' set of coefficients for averaging.

3. Prototype Selection: The power of the synthesis process is that it characterises varying amounts of local structure in a textured image. A suitable texture element plays a key role to synthesis and analysis of the textures under study. Its limitation is that the prototype block is currently selected randomly at each level. The lack of an explicit procedure to isolate a texture element weakens the power of this algorithm. A topic of future research could be to investigate this problem and in particular to develop a procedure within a multiresolution framework for generating a notional texture element at some level which takes the Fourier phase into account, as well as propagating information to the next level for the purpose of deriving a suitable prototype.

4. Texture Element Scale: Because this work uses the MFT, in which scale and frequency are independently varied, the scale of analysis can be automatically adapted to the data. However, the decision of the best scale of the texture element in the present scheme is based on the final synthesis results at each level, instead of measuring the scale locally *during* the synthesis process.

5. Synthesis: The synthesis is carried out by transforming the prototype via the inverse of the identified affine transform and the spectrum is approximated by interpolating that in the target patch. From the view of information coding, a generalisation of the model would allow the necessary information to be encoded

to achieve texture coding. Importantly, no use was made of the 'residual error' in the synthesis, except in defining the signal-noise ratio. In the context of coding, the residuals are usually important.

6.3 Concluding Remarks

This work has proposed an effective approach to modelling textures which contain varying amounts of local structure. Texture boundaries have been detected with various texture types. The proposed texture analysis provides not only a new perspective on textured shape analysis and potential for the coding of textures but also provides a basis for a general approach to texture modelling and analysis with potential applications in many areas of image analysis and computer vision.

Appendix A

Conference Paper

This paper was presented at the 8th. SCIA, Tromso, Norway, 1993 [55].

Texture Analysis using the Multiresolution Fourier Transform

Tao-I. Hsu, A.D. Calway, R. Wilson

Dept. of Computer Science,University of Warwick,
Coventry CV4 7AL, England.

Abstract

A new frequency domain approach to the analysis of texture is described. The method works by identifying, for pairs of texture 'patches' of a given size, the affine co-ordinate transformation which gives the best match between them. This allows the analysis to take account of the geometric warping which is typically found in images of natural textures. The scheme is implemented efficiently in the Fourier domain, and, by variation of scale using the Multiresolution Fourier Transform (MFT), it is possible to identify the smallest scale which gives acceptable results in the matching process. The technique has the potential to deal effectively with textures having varying amounts of structure and can be used both for segmentation and resynthesis of textures from a single prototype block.

1 Introduction

The analysis of textures is one of the oldest problems in image processing, but continues to generate interest and a degree of controversy because of the limited success with which most of the proposed methods of analysis have met [1]. Part of the debate has centred around the issue of *structure* vs. *statistics* : it is widely recognised that many naturally occurring textures exhibit a combination of regularity, such as approximate periodicity, and variation which is hard to model using either straightforward repetition or traditional statistical techniques [1, 2, 3, 7]. In particular, there has been much work based on spectral analysis [4, 5, 6, 7], in which methods have become ever more sophisticated. Yet even when these methods yield good segmentation results, their ability to capture textural properties is limited because they have generally ignored the phase spectrum, which contains much of the information about spatial structure [8]. Moreover, the analysis method described in [7], which used a form of wavelet transform, was unable to cope with slow variation of structure caused by the 3-D shape of the textured surface, which can result in large changes in the local spectrum across regions which are homogeneous. Such variation can be useful in estimating the 3-D structure [9] or in synthesis [10] and ought to be accommodated in any general texture model.

The technique described here is a generalisation of those of [7] to solve the problem of modelling structural properties of textures. As in [7], the starting point is a local Fourier representation, but in the present work this is the multiresolution Fourier transform (MFT), a generalisation of the wavelet and windowed Fourier (Gabor) transforms in which scale and frequency are independently varied, allowing local Fourier analysis to be carried out over a range of scales [11]. This means that the scale of analysis can be adapted to the data, a generalisation of the more familiar scale space concept [12]. From the Fourier representation at a given scale, it is possible to identify the affine co-ordinate transform which warps a given block of texture into an approximation of a second, arbitrarily chosen block [9, 10]. In effect, the method is 'analysis-by-synthesis', since it involves the building of an approximation to the test data from a suitable prototype. After a brief account of the method, results are presented to show its effectiveness in dealing with natural textures containing varying levels of structure. The paper concludes with comments on the potential of the approach to texture analysis and a discussion of possible extensions to it.

2 Analysis and Synthesis of Texture Elements

It is appropriate to begin by first considering the analysis and synthesis of a texture element by the affine transformation of a suitable prototype. As indicated above, this is based on the Fourier spectra of the prototype and texture element of interest. It consists of three main components:

1. Determination of the linear transformation which gives the best fit between the magnitude spectra.

2. Linear transformation of the prototype spectrum to give an approximation to that of the texture element with a phase shift corresponding to the spatial translation between them.

3. Correlation of the texture element and transformed prototype by multiplication of the respective spectra and inverse Fourier transformation, giving a 'goodness of fit' measure and an estimate of the spatial translation.

The basis of the approach is therefore to split the identification of the affine transformation into two parts: determination of its linear component by mapping the shift invariant magnitude spectra; and subsequent determination of the translation component using correlation. This has the dual advantage of being both efficient and automatically providing a certainty measure of the match between the two, thus yielding a simple means of assessing the suitability of the model.

The linear transformation giving the best fit between the magnitude spectra is identified using a representation based on first and second order moments. Due to the inherent symmetry, these are calculated for an appropriately oriented half-plane, and, in the case of the second order moments, along its two principal axes to allow for transformation in both dimensions. Specifically, given a discrete Fourier spectrum $\hat{x}(\vec{\omega}_j)$ corresponding to a prototype or texture element at a given scale,

where $\vec{\omega}_j = [\omega_{j1} \;\; \omega_{j2}]^T$ is the frequency coordinate vector, the procedure is to find the angle θ which minimises the half-plane variance

$$\sigma^2(\theta) = \frac{1}{M(\theta)} \sum_{\vec{\omega}_j \in \Lambda(\theta)} |\hat{x}(\vec{\omega}_j)| \;\; \| \vec{\omega}_j - \vec{\mu}(\theta) \|^2 \tag{1}$$

where $M(\theta) = \sum_{\vec{\omega}_j \in \Lambda(\theta)} |\hat{x}(\vec{\omega}_j)|$, the vector $\vec{\mu}(\theta)$ defines the centroid

$$\vec{\mu}(\theta) = \frac{1}{M(\theta)} \sum_{\vec{\omega}_j \in \Lambda(\theta)} |\hat{x}(\vec{\omega}_j)| \vec{\omega}_j \tag{2}$$

and $\Lambda(\theta)$ denotes the set of half-plane coordinates at an angle θ, ie

$$\omega_{j2} \cos\theta - \omega_{j1} \sin\theta \;\; \geq \;\; 0 \qquad \forall \;\; \vec{\omega}_j \;\in\; \Lambda(\theta) \tag{3}$$

as illustrated in Fig. 1a. At the minimum variance angle, θ_o say, the principal axes with respect to the centroid $\vec{\mu} = \vec{\mu}(\theta_o)$ are then given by the orthonormal eigenvectors $\vec{t}_1$ and $\vec{t}_2$ of the tensor T

$$T = \frac{1}{M(\theta_o)} \sum_{\vec{\omega}_j \in \Lambda(\theta_o)} |\hat{x}(\vec{\omega}_j)| (\vec{\omega}_j - \vec{\mu})(\vec{\omega}_j - \vec{\mu})^T \tag{4}$$

and the variance along each axis by the corresponding eigenvalues, λ_1 and λ_2. This leads to a representation of the magnitude spectrum in terms of three vectors: the centroid $\vec{\mu}$ and the 'principal axes' $\vec{\nu}_1$ and $\vec{\nu}_2$

$$\vec{\nu}_i = \begin{cases} \vec{\mu} + \sqrt{\lambda_i}\, \vec{t}_i & \text{if } \vec{t}_i.\vec{\mu} \geq 0 \\ \vec{\mu} - \sqrt{\lambda_i}\, \vec{t}_i & \text{else} \end{cases} \qquad i = 1, 2 \tag{5}$$

where '.' denotes the dot product. As shown in Fig. 1a, the representation therefore defines a pair of ellipses, symmetrically displaced about the θ_o-line and centred on the centroids of the energy densities on the half-planes, with eccentricity corresponding to the standard deviation of the magnitude spectrum along the principal axes.

Given representations of the above form for both the prototype and texture element, the required linear transformation relating the respective magnitude spectra then corresponds to that relating the two sets of representative vectors. Since there are three vectors in each set, the problem is overdetermined and hence the solution is found using a constrained least-squares approach. Moreover, given the symmetry of the spectra, there are four possible linear transformations which must be considered for any given pair of sets of vectors. If the sets $\{\vec{\mu}^{(1)}, \vec{\nu}_1^{(1)}, \vec{\nu}_2^{(1)}\}$ and $\{\vec{\mu}^{(2)}, \vec{\nu}_1^{(2)}, \vec{\nu}_2^{(2)}\}$ represent the prototype and texture element respectively, then it is readily shown that these correspond to associating the vectors in the following combinations:

$$\begin{array}{l} \{\; (-1)^m \vec{\mu}^{(1)} \,,\; \vec{\mu}^{(2)} \;\;;\;\; (-1)^m \vec{\nu}_1^{(1)} \,,\; \vec{\nu}_1^{(2)} \;\;;\;\; (-1)^m \vec{\nu}_2^{(1)} \,,\; \vec{\nu}_2^{(2)} \;\} \\ \{\; (-1)^m \vec{\mu}^{(1)} \,,\; \vec{\mu}^{(2)} \;\;;\;\; (-1)^m \vec{\nu}_1^{(1)} \,,\; \vec{\nu}_2^{(2)} \;\;;\;\; (-1)^m \vec{\nu}_2^{(1)} \,,\; \vec{\nu}_1^{(2)} \;\} \end{array} \qquad m = 0, 1 \tag{6}$$

For each combination, the 2×2 matrix representing the linear transformation is derived. For example, for the first combination with m=0, the matrix A_1 is found

such that $\vec{\mu}^{(1)} = A_1\vec{\mu}^{(2)}$ and the sum $\sum_{i=1}^{2} \parallel \vec{\nu}_i^{(1)} - A_1\vec{\nu}_i^{(2)} \parallel^2$ is a minimum. This gives a least squares solution with the constraint that the two centroids $\vec{\mu}^{(1)}$ and $\vec{\mu}^{(2)}$ are mapped onto each other. The matrices $A_2 - A_4$ corresponding to the remaining three combinations are derived in a similar manner.

The above procedure assumes that the magnitude spectra are characterised by a single cluster of energy within an appropriate half-plane, ie that there is a unique minimum of the variance $\sigma^2(\theta)$ with respect to the orientation θ. In the case of more than one minimum, however, the representation is potentially ambiguous. This is readily seen from Fig. 1b, in which two equally valid representations are clearly not compatible with a linear transformation. This ambiguity can be overcome by using an alternative representation in terms of two centroid vectors defining the centre of each of two energy clusters as illustrated in Fig. 1c. If the two spectra are related via a linear transformation, then this will correspond to that relating the two centroid vectors - the existence of two separated clusters enables the problem to be further constrained and hence yields an exact solution. Thus, for a given spectrum $\hat{x}(\vec{\omega}_j)$, the angles θ_1 and θ_2 are sought such that the sum of the variances

$$\sigma_i^2(\theta_1,\theta_2) = \frac{1}{M_i(\theta_1,\theta_2)} \sum_{\vec{\omega}_j \in \Lambda_i(\theta_1,\theta_2)} |\hat{x}(\vec{\omega}_j)| \parallel \vec{\omega}_j - \vec{\mu}_i(\theta_1,\theta_2) \parallel^2 \qquad i = 1,2 \tag{7}$$

is a minimum, where

$$M_i(\theta_1,\theta_2) = \sum_{\vec{\omega}_j \in \Lambda_i(\theta_1,\theta_2)} |\hat{x}(\vec{\omega}_j)| \tag{8}$$

$\vec{\mu}_i(\theta_1,\theta_2)$ is the centroid

$$\vec{\mu}_i(\theta_1,\theta_2) = \sum_{\vec{\omega}_j \in \Lambda_i(\theta_1,\theta_2)} |\hat{x}(\vec{\omega}_j)| \, \vec{\omega}_j \tag{9}$$

and $\Lambda_1(\theta_1,\theta_2)$ and $\Lambda_2(\theta_1,\theta_2)$ are the sets of coordinates in each segment of the half-plane at an angle θ_1 and divided at an angle θ_2 as illustrated in Fig. 1c, ie

$$\Lambda_1(\theta_1,\theta_2) = \Lambda(\theta_1) - \Lambda(\theta_1+\theta_2) \cap \Lambda(\theta_1) \tag{10}$$

$$\Lambda_2(\theta_1,\theta_2) = \Lambda(\theta_1) \cap \Lambda(\theta_1+\theta_2) \tag{11}$$

where $\Lambda(\theta)$ is as defined in eqn. (3). The centroids of the segments corresponding to the minimum variance sum, $\vec{\mu}_1$ and $\vec{\mu}_2$, are then the required vectors. In this case, the symmetry of the spectra means that there are eight possible linear transformations that need to be tested (two each for a given vector and its negative). This in turn leads to the 2×2 matrices $A_5 - A_{12}$. For example, the matrix A_5 is defined such that $\vec{\mu}_i^{(1)} = A_5\vec{\mu}_i^{(2)}$, $i = 1,2$.

Using both of the above matching procedures, the twelve possible transformations linking the magnitude spectra based on the single and double cluster models are computed. To determine which of these gives the best fit, the 'synthesis followed by correlation' method is adopted: for each case, the prototype is transformed and correlated with the original texture element to find the best match. Thus, for each of the matrices $A_1 - A_{12}$, the prototype spectrum $\hat{x}^{(1)}(\vec{\omega}_j)$ is transformed according to

$$\hat{x}_i^{(1)}(\vec{\omega}_j) = \hat{x}^{(1)}(A_i\vec{\omega}_j) \qquad 1 \le i \le 12 \tag{12}$$

where a given coefficient $\hat{x}_i^{(1)}(\vec{\omega}_j)$ is calculated from the discrete coefficients $\hat{x}^{(1)}(\vec{\omega}_j)$ using bilinear interpolation from its four nearest neighbours. The transformed prototypes are then correlated with the texture element using [13]

$$\rho_i(\vec{\xi}_j) = F^{-1}\{\hat{x}_i^{(1)}(\vec{\omega}_j)\hat{x}^{(2)^*}(\vec{\omega}_j)\} \tag{13}$$

where * is the complex conjugate, F^{-1} denotes the inverse discrete Fourier transform (DFT), and $\vec{\xi}_j$ is the spatial coordinate vector. The correlation with the largest peak then determines the selected transformation, ie the best match is achieved by transforming the prototype according to the matrix A_k such that

$$\epsilon = \max_j \rho_k(\vec{\xi}_j) = \max_i (\max_j \rho_i(\vec{\xi}_j)) \tag{14}$$

where ϵ is the 'goodness of fit' measure. Finally, the synthesized texture element is then given by

$$x_k^{(1)}(\vec{\xi}_j) = F^{-1}\{\hat{x}_k^{(1)}(\vec{\omega}_j)e^{j\vec{\omega}_j.\vec{\xi}_l}\} \tag{15}$$

where $j^2 = -1$ and $\vec{\xi}_l$ is the translation between the transformed prototype and the texture element, ie $\vec{\xi}_l : \rho_k(\vec{\xi}_l) = \max_j \rho_k(\vec{\xi}_j)$.

3 Implementation and Experiments

Central to implementing the matching scheme is the availability of spatially localised spectra for the prototype and texture element. Moreover, since the most appropriate scale will be unknown *a priori*, ie the size of the ideal prototype will vary according to the structure within a given texture, the spectra will need to be available over a range of scales. As indicated earlier, these are obtained from the MFT, which, for the discrete image $x(\vec{\xi}_j)$, is given by [11]

$$\hat{x}(\vec{\xi}_i(n), \vec{\omega}_j(n), n) = \sum_k w_n(\vec{\xi}_k - \vec{\xi}_i(n))x(\vec{\xi}_k)\exp[-j\vec{\omega}_j(n).\vec{\xi}_k] \tag{16}$$

and hence corresponds to the DFT of the region defined by the window $w_n(\vec{\xi}_j)$ centred on the point $\vec{\xi}_i(n)$, where the spatial extent of the former varies with the scale index n. In the present work, the region centres were arranged on a uniform cartesian lattice with intervals reflecting the window size on each level and an effective overlapp of 50% to allow for interaction between adjacent regions. Further details of this and those relating to the frequency sampling and window functions used can be found elsewhere [11].

To test the performance of the matching algorithm, resynthesis experiments based on natural images were performed, in which a given texture was generated from an arbitrarily chosen 'prototype block' at various scales. Using the appropriate local spectra, each texture block was synthesized from the prototype block using the procedure detailed above and the blocks combined to give the final synthesis. A division of the frequency domain into angular segments with associated centroid and mass enabled fast identification of the representative vectors and inverse FFTs provided

for efficient correlation. Results of these experiments for three different textures and at different scales are shown in Fig. 2. Table 1 lists the effective region size at each scale and the corresponding SNR of the original to the synthesized texture. Note that the use of smooth, overlapping windows has prevented blocking artefacts. Moreover, the scale at which a given SNR is achieved can be seen from Table 1 to depend on the regularity of the texture, as might be expected.

4 Conclusions

It has been shown that it is possible to model textures containing varying degrees of local structure by the use of affine transformation of a single prototype pattern. The accuracy of the reconstructions obtained using this technique demonstrates that the method captures the important features of the textures studied. Moreover, it is computationally efficient, requiring little more than one forward and one inverse FFT for each scale examined. It therefore has the potential for segmentation and synthesis applications. These applications are currently being investigated.

Acknowledgement: This work was supported by the Government of the Republic of China in Taiwan.

References

[1] R.M. Haralick, "Statistical and structural approaches to texture", *IEEE Proc.*, 67, pp. 610-21,1979.

[2] G.R. Cross, A.K. Jain, "Markov random field texture models", *IEEE Trans. Patt. Anal. Machine Intell.*, PAMI-5, pp. 25-39, 1983.

[3] S. Zucker, "Towards a model of texture", *Comput. Graph. Image Proc.*, 5, pp. 190-202, 1976.

[4] J.S. Weszka, C. Dyer, A. Rosenfeld, "A comparative study of texture measures for terrain classification", *IEEE Trans. Sys. Man Cybern.*, SMC-6, 1976.

[5] H.E. Knutsson, G.H. Granlund, "Texture analysis using 2-D quadrature filters", *Proc. IEEE CAPADIM Workshop*, Pasadena, 1983.

[6] A.C. Bovik, M. Clark, W.S. Geisler, "Multichannel texture analysis using localized spatial filters", *IEEE Trans. Patt. Anal. Machine Intell.*, PAMI-12, pp. 55-73, 1990.

[7] R. Wilson, M. Spann, *Image Segmentation and Uncertainty*, Letchworth, Res. Studies Pr., 1988.

[8] A.V. Oppenheim, J.S. Lim, "The importance of phase in signals", *IEEE Proc.*, 69, pp. 529-41, 1981.

[9] J. Gårding, *Shape from Surface Markings*, Ph.D. Thesis, Royal Inst. of Technology, Stockholm, 1991.

[10] P. Volet, M. Kunt, "Synthesis of natural structured textures", *Proc. EUSIPCO-86*, pp. 913-6, The Hague, 1986.

[11] R. Wilson, A.D. Calway, E.R.S. Pearson,"A generalized wavelet transform for Fourier analysis: the multiresolution Fourier transform and its application to image and audio signal analysis", *IEEE Trans. Inf. Th.*, IT-38, pp. 674-90, 1992.

[12] A. Witkin, "Scale-Space Filtering", *Proc. IEEE ICASSP-84*, San Diego, 1984.

[13] A.Papoulis, *Signal Analysis*, New York, McGraw-Hill, 1977.

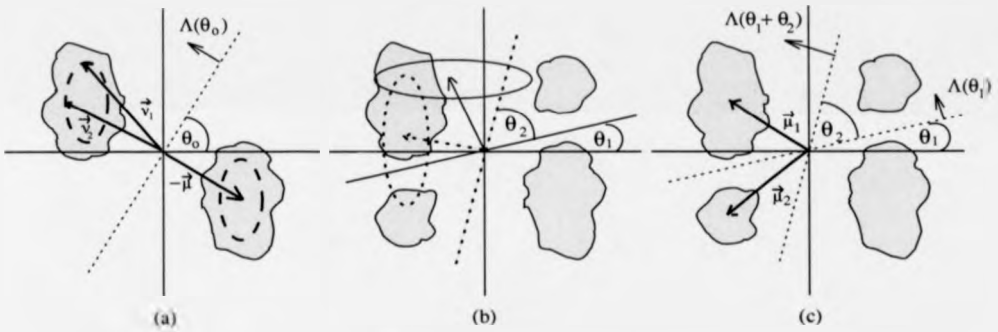
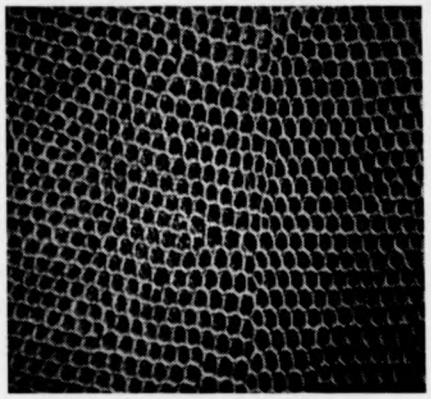


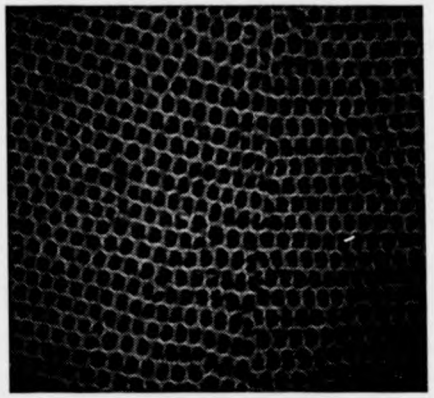
Figure 1: (a) Vectors $\vec{\mu}$, $\vec{\nu}_1$, and $\vec{\nu}_2$ define the centroid and principal axes of the ellipse representing the magnitude spectrum for the minimum variance half-plane at angle θ_o with coordinates $\Lambda(\theta_o)$. (b) The representation of (a) is ambiguous for multiple clusters: equally valid representations for variance minima at angles θ_1 and θ_2 are not related via a linear transformation. (c) Ambiguity in (b) is overcome by seeking divisions at angles θ_1 and θ_2, and using centroid vectors $\vec{\mu}_1$ and $\vec{\mu}_2$.

MFT Scale Index	Effective Region Size (pixels)	SNR(dB)		
		Reptile	Burlap	Grass
3	8 × 8	8.64229	4.04274	4.06653
4	16 × 16	7.11266	3.72840	2.07278
5	32 × 32	4.76636	1.79312	-0.42035
6	64 × 64	2.98247	-0.21225	-1.64412

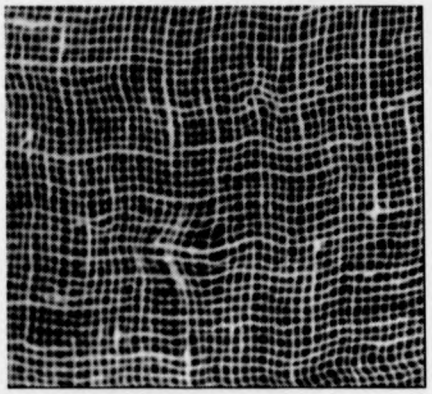
Table 1: SNR of original to synthesized texture for various scales.



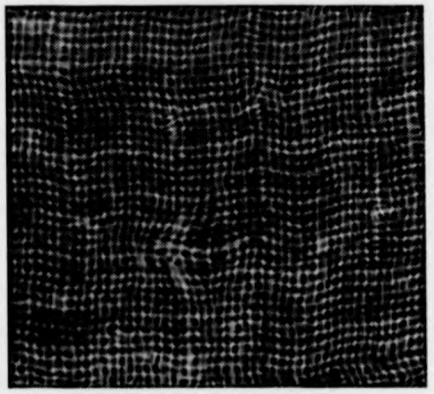
(a) Original 'Reptile' texture.



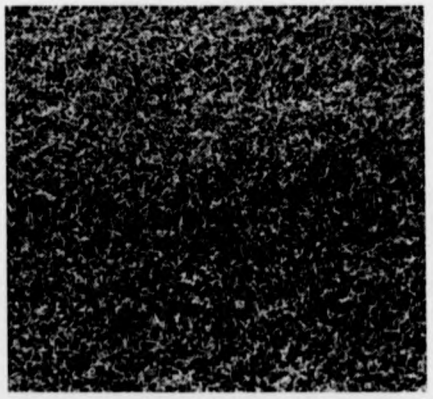
(b) Synthesis at scale index 5.



(c) Original 'Burlap' texture.



(d) Synthesis at scale index 4.



(e) Original 'Grass' texture.

(f) Synthesis at scale index 3.

Figure 2: Single block synthesis of natural textures at various scales (cf Table 1)

Appendix B

Conference Paper

This paper is accepted for publish in the 1^{th}. IEEE International Conference on Image Processing, Austin, Texas, U.S.A., November, 1994 [56].

PUBLISHED PAPERS NOT FILMED FOR COPYRIGHT REASONS

P162 – P168

Bibliography

[1] C. O. Acuna. Texture Modeling Using Gibbs Distributions. *CVGIP: Graphical Models and Image Processing*, 54:210–222, 1992.

[2] A.Khotanzad and J. Chen. Unsupervised Segmentation of Textured Images by Edge Detection in Multidimensional Features. *IEEE Trans. Patt. Anal. Machine Intell.*, 4:414–421, April 1989.

[3] J. Allen and L. R. Rabiner. A Unified Approach to Short Time Fourier Analysis and Synthesis. *IEEE*, 65:1558–1564, 1980.

[4] R. Bajcsy. Computer Description of Textured Surfaces. In *Proc. of 3rd Int. Joint Conf. Art. Int.*, pages 572–579, August 1973.

[5] R. Bajcsy and L. Lieberman. Texture Gradient as depth cue . *Computer Vision, Graphics and Image Processing*, 5:52–67, March 1976.

[6] M. T. Barnsley. *Fractals Everywhere.* Academic Press., New York, 1988.

[7] A. H. Bhalerao. *Multiresolution Image Segmentation.* PhD thesis, The University of Warwick, UK, 1991.

[8] C. Bouman and B. Liu. Multiple Resolution Segmentation of Textured Images. *IEEE Trans. Patt. Anal. Machine Intell.*, 13:99–113, February 1991.

[9] A. Bovik, M. Clark, and W. Geisler. Multichannel Texture Analysis Using Localized Spatial Filters. *IEEE Trans. Patt. Anal. Machine Intell.*, 12(1):55–72, January 1990.

[10] R. N. Bracewell. *The Fourier Transform and Its Applications.* McGraw Hill Book Company, second edition, 1986.

[11] P. Brodatz. *A Photographic Album for Artists and Designers.* Dover Publishing Company, 1966.

[12] P. J. Burt. Fast Filter Transform for Image Processing. *Computer Graphic Image Processing*, 16:20–51, 1981.

[13] P. J. Burt. Pyramid Based Extraction of Local Image Features with Applications to Motion and Texture Analysis. In *Soc. Photo-Opt. Instrum. Eng.*, pages 114–124, San Diego, 1982.

[14] P. J. Burt and E. H. Adelson. The Laplacian Pyramid as a Compact Image Code. *IEEE Trans. Commun.*, COM(31):532–540, 1983.

[15] T. Caelli. On Discriminating Visual Textures and Images. *Perception and Psycophysics*, 31:149–159, 1982.

[16] A. D. Calway. *The Multiresolution Fourier Transform: A general Purpose Tool for Image Analysis.* PhD thesis, Department of Computer Science, The University of Warwick, UK, September 1989.

[17] R. Chellappa and S. Chatterjee. Texture Classification using GMRF Models. *IEEE Trans. Acous. Speech Sig. Proc.*, 33:959–963, August 1985.

[18] R. Chellappa and R. L. Kashyap. Texture Synthesis using 2-D Noncausal Autoregressive Models. *IEEE Trans. Acous. Speech Sig. Proc.*, 33:194–203, August 1985.

[19] C.H. Chen. A Study of Texture Classifications using Spectral Features. In *6th Int. Conference on Pattern Recognition*, pages 1064–1067, Munich, 1982.

[20] S. C. Clippingdale and R. Wilson. Quad-Tree Image Estimation: A New Image Model and its Application to MMSE Image Restoration. In *Proc. 5th Scandinavian Conf. Image Analysis*, pages 699–706, Stockholm, Sweden, 1987.

[21] F. S. Cohen and D. B. Cooper. Simple parallel Hierarchical and Relaxation Algorithms for Segmenting Noncausal Markov Random Fields. *IEEE Trans. Patt. Anal. Machine Intell.*, 9:195–219, 1987.

[22] F. S. Cohen and Z. Fan. Maximum Likelihood Unsupervised Textured Image Segmentation. *CVGIP:Graphical Models and Image Processing*, 54:239–251, 1992.

[23] R. W. Conners and C. A. Harlow. A Theoretical Comparison of Texture Algorithms. *IEEE Trans. Patt. Anal. Machine Intell.*, 2(3):204–222, May 1980.

[24] R. E. Crochiere and L. R. Rabiner. Optimum FIR digital Filter Implementation for Discrimination, Interpolation and Narrow Band Filtering. *IEEE Trans. Acous. Speech Sig. Proc.*, ASSP-23(5):444–464, January 1975.

[25] G.R. Cross and A.K. Jain. Markov Random Field Texture Models. *IEEE Trans. Patt. Anal. Machine Intell.*, 5:25–39, 1983.

[26] L. Daubechies. Orthogonal Bases of Compactly Supported Wavelets. *Comm. on Pure and Appl. Math.*, XLI:909–996, 1988.

[27] L. Daubechies. The Wavelet Transform, Time-Frequency Localisation and Signal Analysis. *IEEE Trans. Information Theory*, 36:961–1005, 1990.

[28] J. G. Daugman. Uncertainty Relation for Resolution in Space, Spatial Frequency and Orientation Optimized by Two-dimensional Vision Cortical Filters. *J. Opt. Soc. Amer. A*, 2(7):1160–1169, July 1985.

[29] J. G. Daugman. An Information-theoretic View of Analog Representation in Striate Cortex. In E.L. Schwarz, editor, *Computational Neuro-science*. MIT Press, 1988.

[30] A. R. Davies. *Image Feature Analysis Using the Multiresolution Fourier Transform*. PhD thesis, Department of Computer Science, The University of Warwick, UK, August 1993.

[31] H. Derin and H. Elliott. Modeling and Segmentation of Noisy and Textured Images Using Gibbs Random Fields. *IEEE Trans. Patt. Anal. Machine Intell.*, 9:39–55, 1987.

[32] J. M. H. du Buf. Abstract Processes in Texture Discrimination. *Spatial Vision*, 6(3):221–242, 1992.

[33] J. M. H. du Buf, M. Kardan, and M. Spann. Texture Feature Performance for Image Segmentation. *PR*, 23:291–308, 1990.

[34] D. Dunn, W. Higgins, and J. Wakeley. Texture Segmentation Using 2-D Gabor Elementary Functions. *IEEE Trans. Patt. Anal. Machine Intell.*, 16(2):130–149, February 1994.

[35] J. Eklundh. On the Use of Fourier Phase Features for Texture Discrimination. *Computer Graph. Image Proc.*, 9:199–201, 1979.

[36] I. M. Elfadel and R. W. Picard. Gibbs Random Fields, Cooccurrences, and Texture Modeling. *IEEE Trans. Patt. Anal. Machine Intell.*, 16:24–37, 1994.

[37] O.D. Faugeras. Texture Analysis and Classification using a Human Visual Model. In *Proc. of Int. Conf. Pattern Recogn.*, pages 7–10, Kyoto,Japan, November 1978.

[38] J. M. Francos, A. Meiri, and B. Porat. A Unified Texture Model Based on a 2-D Wold-Like Decomposition. *IEEE Trans. Signal Processing*, 41:2665–2677, 1993.

[39] K. Fukunaga. *Introduction to Statistical Pattern Recognition*. New York, Academic Press Ltd, 1979.

[40] D. Gabor. Theory of Communication. *Proc. IEE*, 93(26):429–441, November 1946.

[41] A. Gagalowicz. A New Method for Texture Fields Synthesis, Some Applications to the Study of Human Vision. *IEEE Trans. Patt. Anal. Machine Intell.*, 3(5), September 1981.

[42] M. M. Galloway. Texture Analysis using Gray Level Run Length. *Computer Graphics Image Processing*, 4:172–179, 1975.

[43] J. Gårding. *Shape from Surface Markings*. PhD thesis, Inst. of Technology, Stockholm, 1991.

[44] D. Geman, S. Geman, C. Graffigne, and P. Dong. Boundary Detection by Constrained Optimisation. *IEEE Trans. Patt. Anal. Machine Intell.*, 12:609–28, 1990.

[45] S. Geman and D. Geman. Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images. *IEEE Trans. Patt. Anal. Machine Intell.*, 6(6):721–741, 1984.

[46] L.Van Gool, P. Dewaele, and A. Oosterlinck. Texture Analysis Anno 1983. *Computer Vision, Graphics and Image Processing*, 29:336–357, 1985.

[47] N. Graham and B. Rogowitz. Spatial Pooling Properties Deduced from the Detectability of FM and Quasi-AM Grattings: A reanalysis. *Vision Research*, 16:1021–1026, 1976.

[48] G. H. Granlund. In Search of a General Picture Processing Operator. *Computer Graphics Image Processing*, 8:155–173, 1978.

[49] A. Grossman and J. Mortlet. Decomposition of Hardy functions into Square Integrable Wavelengths of Constant Shape. *SIAM J. Math. Anal.*, 15:723–736, 1984.

[50] J. Haddon and J. Boyce. Image Segmentation by Unifying Region and Boundary Information. *IEEE Trans. Patt. Anal. Machine Intell.*, 12(10):929–948, October 1990.

[51] R. Haralick. Statistical and Structural Approaches to Texture. *IEEE*, 67(5):610–621, 1979.

[52] R. M. Haralick and L. Shapiro. *Computer and Robot Vision*. Reading, Mass., London, Addison-Wesley, 1993.

[53] R.M. Haralick. Statistical Image Texture Analysis. In T.Y. Young and K. S. Fu, editors, *Handbook of Pattern Recognition and Image Processing*. Academic Press Inc., 1985.

[54] J.K. Hawkins. Textural Properties for Pattern Recognition. In B.S. Lipkin and A. Rosenfeld, editors, *Picture Processing and Psycophysics*. New York Academic Press, 1970.

[55] Tao-I Hsu, A D Calway, and R Wilson. Texture analysis using the multiresolution fourier transform. In *8th Scandinavian Conference on Image Analysis*, pages 823–830, Tromso, Norway, May 1993.

[56] Tao-I Hsu and R Wilson. Texture analysis using a generalised wavelet transform. In *IEEE Inter. Conf. on Image Processing*, page accepted, Austin, Texas, USA, November 1994.

[57] A. Jacquin. Image coding based on a fractal theory of iterated contractive image transformations. *IEEE Trans. Image Processing*, 1:18–30, 1991.

[58] A. K. Jain. Advances in Mathematical Models for Image Processing. *Proc. IEEE*, 69:502–528, 1981.

[59] A. K. Jain and F. Farrokhnia. Unsupervised Texture Segmentation Using Gabor Filter. *Pattern Recognition*, 24(12):1167–1186, 1991.

[60] B. Julesz. Experiments in the Visual Perception of Texture. *Science Amer.*, 232:34–43, 1975.

[61] B. Julesz and J. Bergen. Textons, the Fundamental Elements in Preattentive Vision and Perception of Textures. *Bell Syst. Tech J.*, 62:1619–1645, 1983.

[62] D. H. Kelly. Spatial Frequency Selectivity in the Retina. *Vision Research*, 15:665–672, 1975.

[63] H. Knutsson. *Filtering and Reconstruction in Image Processing.* PhD thesis, University of Linköping, Sweden, 1982.

[64] J.J. Koenderink. The Structure of Images. *Biol. Cybern.*, 50:363–370, 1984.

[65] L.I Larkin and P.J.Burt. Multiresolution texture energy measures. In *IEEE CVPR*, pages 512–520, Washington D.C., 1983.

[66] K. I. Laws. *Textured Image Segmentation.* PhD thesis, Department of Engineering, University of Southern California, 1980.

[67] S. G. Mallat. A Theory for Multiresolution Signal Decomposition: The Wavelet Representation. *IEEE Trans. Patt. Anal. Machine Intell.*, 11(7):674–693, July 1989.

[68] S. G. Mallat. Multifrequency Channel Decompositions of Images and Wavelet Models. *IEEE Trans. Acous. Speech Sig. Proc.*, 37(12):2091–2110, December 1989.

[69] J. Mao and A. Jain. Texture Classification and Segmentation using Multiresolution Simultaneous Autoregressive Models. *Pattern Recognition*, 25(2):173–188, 1992.

[70] D. Marr. *Vision.* San Francisco, Freeman, 1980.

[71] T. Matsuyama, S. Miura, and M. Nagao. A Structural Analysis of Natural Textures by Fourier Transformation. In *6th Proc. of Int. Conf. Pattern Recog.*, pages 289–292, Munich, 1982.

[72] R. W. McColl. *Colour Image Quantisation and Coding for Optimal Perception.* PhD thesis, Department of Computer Science, University of Warwick, 1991.

[73] G. Nicholls and M. Petrou. A generalization of renormalization-group methods for multiresolution image-analysis. In *11th International Conference on Pattern Recognition*, pages 567–570, Hague Netherlands, 1992.

[74] A. V. Oppenheim and J. Lim. The Importance of Phase in Signals. *Proceedings IEEE*, 69(5):529–541, May 1981.

[75] A. Papoulis. *Signal Analysis.* McGraw-Hill, 1984.

[76] A. Papoulis and M. S. Bertran. Digital Filtering and Prolate Function. *IEEE Trans. Circuit Theory*, CT-19:674–681, 1972.

[77] E. R. S. Pearson. *The Multiresolution Fourier Transform and its Application to Polyphonic Audio Analysis.* PhD thesis, Department of Computer Science, The University of Warwick, UK, September 1991.

[78] S. Peleg, J. Naor, R. Hartley, and D. Avnir. Multiple Resolution Texture Analysis and Classification. *IEEE Trans. Patt. Anal. Machine Intell.*, 6:518–523, 1984.

[79] M. Pietikainen and A. Rosenfeld. Image Segmentation by Texture using Pyramid Node Linking. *IEEE Trans. Sys. Man Cyber.*, SMC-11:822–825, 1981.

[80] M. Pietikainen and A. Rosenfeld. Image Segmentation by Texture using Pyramid Node Linking. *IEEE Trans. Sys. Man Cyber.*, SMC-11:822–825, 1981.

[81] M. R. Portnoff. Time-frequency Representation of Digital Signals and Systems B ased on Short-Time Fourier Analysis. *IEEE Trans. Acous. Speech Sig. Proc.*, 28(1):55–69, February 1980.

[82] W. Pratt. *Digital Image Processing.* John Wiley & Sons, Inc., 1976.

[83] T. R. Reed and J. M. H. du Buf. A Review of Recent Texture Segmentation and Feature Extraction Techniques. *Image Understanding*, 57:359–372, 1993.

[84] S. Roan and J. Aggarwal. Multiple Resolution Imagery and Texture Analysis. *IEEE Trans. Patt. Anal. Machine Intell.*, 20(1):17–31, 1987.

[85] A. Rosenfeld. *Multiresolution Image Processing and Analysis.* Springer-Verlag, 1982.

[86] A. Rosenfeld, R. A. Hummel, and S. W. Zucker. Scene Labelling by Relaxation Operations. *IEEE Trans. Sys. Man Cyber.*, SMC-6:420–433, 1976.

[87] R. Sekuler. Spatial Vision. *Annual Review of Psychology*, 25:195–232, 1974.

[88] C. E. Shannon. A Mathematical Theory of Communication. *The Bell Sys. Tech. J.*, 27:379–423, 1948.

[89] D. Slepian and H. O. Pollak. Prolate Spheroidal Wavefunctions, Fourier Analysis and Uncertainty I. *Bell Syst. Tech. J.*, 40:43–64, 1961.

[90] M. Spann. *Texture Description and Segmentation in Image Processing.* PhD thesis, Department of Electrical and Electronic Engineering, University of Aston in Birmingham, 1985.

[91] M. Spann and R. G. Wilson. A Quad-Tree Approach to Image Segmentation Which Combines Statistical and Spatial Information. *Pattern Recognition*, 18(3/4):257–269, 1985.

[92] T. Tan. Texture Edge Detection Based on A Visual Cortical Channel Model. In *Proc. of The 8th Scandinavian Conf. on Image Analysis*, pages 815–821, 1993.

[93] D. Thomson. Spectrum Estimation and Harmonic Analysis. *IEEE.*, 70:1055–1096, 1982.

[94] D. W. Tufts and J. T. Francis. Designing Digital Filters-Comparison of Methods and Criteria. *IEEE Trans Audio Electroacoust.*, 18:487–494, 1970.

[95] M. Unser and M. Eden. Multiresolution Feature Extraction and Selection for Texture Segmentation. *IEEE Trans. Patt. Anal. Machine Intell.*, 11:717–728, 1989.

[96] E. Vilnratter, R. Nevatia, and K. Price. Structural Analysis of Natural Textures. *IEEE Trans. Patt. Anal. Machine Intell.*, 8:76–89, 1986.

[97] P. Volet and M. Kunt. Synthesis of Natural Structured Textures. In *Proc. of EUSIPCO-86*, pages 913–916, 1986.

[98] D. Wermser and C. Liedtke. Texture Gradient: A new Tool for the Unsupervised Segmentation. In *The Third Scandinavian Conference on Image Analysis*, pages 85–89, July 1983.

[99] J.S. Weszka, C.R. Dyer, and A. Rosenfeld. A Comparative Study of Texture Measures for Terrain Classification. *IEEE Trans. Sys. Man Cyber.*, 6(4):269–285, April 1976.

[100] R. Wilson. British Machine Vision Conference 93 Tutorial Notes on Wavelet Transforms. *BMVC 93 Tutorial Notes*, 1993.

[101] R. Wilson and A. H. Bhalerao. Kernel Designs for Efficient Multiresolution Edge Detection an d Orientation Estimation. *IEEE Trans. Patt. Anal. Machine Intell.*, 14:384–390, 1992.

[102] R. Wilson, A. Calway, and E.R.S. Pearson. A Generalized Wavelet Transform for Fourier Analysis: The Multiresolution Fourier Transform and Its Application to Image and Audio Signal Analysis. *IEEE Trans. Information Theory*, 38(2):674–690, March 1992.

[103] R. Wilson, A. Calway, E.R.S. Pearson, and A. Davies. An Introduction of Multiresolution Fourier Transform and it's applications. Research Report RR204, Department of Computer Science, University of Warwick, UK, January 1992.

[104] R. Wilson and M. Spann. Finite Prolate Spheroidial Sequence and Their Applications II: Image Feature Description and Segmentation. *IEEE Trans. Patt. Anal. Machine Intell.*, 10:193–203, 1988.

[105] R. Wilson and M. Spann. A New Approach To Clustering. *Pattern Recognition*, 23(12):1413–1425, 1990.

[106] R. G. Wilson. Finite Prolate Spheroidal Sequences and Their Applications I: Generation and Properties. *IEEE Trans. Patt. Anal. Machine Intell.*, 9(6):787–795, November 1987.

[107] R. G. Wilson, H. Knutsson, and G. H. Granlund. Anisotropic Nonstationary Image Estimation and its Applications II - Predictive Image Coding. *IEEE Trans. Commun.*, COM-31:398–406, 1983.

[108] R. G. Wilson and M. Spann. *Image Segmentation and Uncertainty*. Pattern Recognition and Image Processing Series. Research Studies Press Ltd, 1988.

[109] A. Witkin. Scale-Space Filtering. In *Proc. of IEEE ICASSP-84*, 1984.

[110] C. M. Wu and Y. C. Chen. Statistical Feature Matrix for Texture Analysis. *CVGIP:Graphical Models and Image Processing*, 54:407–419, 1992.

[111] H-C. Yang and R. Wilson. Orientation-directed edge detection using a Hopfield neural network. In *Proc. 8th IEEE workshop on Image and Multidimensional Signal Processing*, Cannes, France, 1993.

[112] S. W. Zucker. Finding Structure in Cooccurrence Matrices for Texture Analysis. *Computer Graphics Image Processing*, 12:286–308, 1980.

TITLE TEXTURE ANALYSIS AND SYNTHESIS USING THE MULTIRESOLUTION FOURIER TRANSFORM.

AUTHOR Tao I HSU

DEGREE Ph.D

AWARDING BODY Warwick University

DATE 1994

THESIS NUMBER DX187297